PB-246 847

DOCUMENTATION FOR ES001. A STEADY-STATE, ONE DIMENSIONAL, ESTUARINE WATER QUALITY MODEL

Steven C. Chapra, et al

Environmental Protection Agency
New York, New York

September 1973

DISTRIBUTED BY:

NTIS
National Technical Information Service
U. S. DEPARTMENT OF COMMERCE

328254

PB 246 847

UNITED STATES ENVIRONMENTAL PROTECTION AGENCY
REGION II
26 FEDERAL PLAZA
NEW YORK, NEW YORK 10007

Documentation for

ES001

"A Steady-state, One Dimensional, Estuarine Water Quality Model"

Steven C. Chapra, Sanitary Engineer
Systems Analysis Section
Seymour Gordimer, Chief
Computer Systems Section
Data Systems Branch

September, 1973

Reproduced by
NATIONAL TECHNICAL
INFORMATION SERVICE
US Department of Commerce
Springfield, VA. 22151

BIBLIOGRAPHIC DATA SHEET	1. Report No. ES001	2.	3. Recipient's Accession No. PB 246 847

4. Title and Subtitle
ES001: A Steady-state, One Dimensional, Estuarine Water Quality Model

5. Report Date
September 1973

6.

7. Author(s)
Chapra, Steven C. and Gordimer, Seymour

8. Performing Organization Rept. No.

9. Performing Organization Name and Address
United States Environmental Protection Agency/Region 2
Data Systems Branch
26 Federal Branch
New York, N.Y. 10007

10. Project/Task/Work Unit No.

11. Contract/Grant No.

12. Sponsoring Organization Name and Address

13. Type of Report & Period Covered
FINAL

14.

15. Supplementary Notes

16. Abstracts

ES001 is a computer program which can be used to simulate tidally averaged one dimensional steady-state estuarine systems and predict water mathematical model of an estuarine system. The basic approach used in ES001 is the law of conservation of mass.

The model presented in this report is specifically applied to the BOD-DO. deficit system, however analogous systems of sequential reactions of two substances, or single systems under first order kinetics may also be modelled.

17. Key Words and Document Analysis. 17a. Descriptors

*Mathematical Model, *Water Quality, *Computer Model, Simulation Analysis, Oxygen Demand, Nutrients.

17b. Identifiers/Open-Ended Terms

17c. COSATI Field/Group

18. Availability Statement Release to Public	19. Security Class (This Report) UNCLASSIFIED	21. No. of Pages 224
	20. Security Class (This Page UNCLASSIFIED	22. Price 7 75

Table of Contents

Page

Introduction 2

Theory 5

  Basic Theory 7
  Development of Water Quality Equations for an Estuary. 12
  Other Applications 28
  Segmentation of Estuary Systems 29
  Evaluation of Boundary Conditions 31
  Sample Problem. 41
  Method for Evaluation and Presentation of Boundary Conditions for Complex Estuaries 47
  Nomenclature 64

Sensitivity. 67

Description of the Computer Program 80

  Description of Main Program with Flowcharts 81
  Description of Subroutines 88
  Description of Functions 100

User's Manual 104

  Initial Steps in Formulating ES001 104
  Discussion of System Parameters 106
  Program Run Preparation 109

Page

Data Description and Input Requirements 113
Output. 124
Restrictions 128
Example Runs 130

References 159

Appendix

ES001 source listing 161
Requirements for running program on IBM 1130 (ES002). 194
Source listing for ES002 195

INTRODUCTION

In 1968, Hydroscience, Inc. prepared a report for the FWPCA entitled, "Mathematical Models for Water Quality for the Hudson-Champlain and Metropolitan Coastal Water Pollution Control Project."[1] The report presented the development procedures of water quality models for various waterways including the New York Harbor Complex, and parts of the Raritan and Hudson Rivers. As part of the project, several computer programs were developed to model the various water systems. Unfortunately, inadequate documentation was available for these programs and an effort has been made by the Environmental Protection Agency to publish such documentation as well as to modify the programs for efficiency and ease of use. It is the purpose of this report to document one of the programs - ES001/2.

ES001 is presently programmed for use on the IBM 370 while ES002 is compatible with an IBM 1130 system. Thus, the designation ES001/2 is used when referring to both programs. In general, the two programs have identical input, output and theoretical basis. The remainder of this documentation, unless explicitly stated, will deal with ES001 in particular with most of the information relevant to ES002.*

In its present form, ES001 is capable of modelling general one-dimensional estuarine systems for a variety of substance concentrations. The program requires segmentation of the system being modelled into sections within which the various geomorphological, physical and hydrological parameters of the estuary are constant. The sections are then connected mathematically, the junction points of these segments being boundary points where these parameters can change. Several types of junctions are allowed including triple junctions, dams, etc., which in combination allow the modelling of numerous types of configurations.

The model is assumed to be at steady state and to be tidally averaged. Steady state means that the variables and parameters describing a system do not vary with time. Investigation of steady-state concentrations is a useful planning strategy for many situations. For example, in summer months, usually the most critical period as far as some types of pollution are concerned, the hydrograph of fresh water flow entering an estuary may be low and relatively constant creating an excellent opportunity for application of this model. Tidally averaged means that fluctuations in response caused by tidal ascillations are not treated explicitly in the model. The effects of tidal mixing are manifested indirectly through inclusion in a bulk dispersion term.

Finally, reaction kinetics are assumed to be of the first-order for the reasons that many substances reacting in an estuary approximate this type of behavior.

---

*An addendum to this report includes the source deck for ES002 and a description of its specifications.

This report assumes that the user has a basic knowledge of stream and estuarine analysis. The basic approach used in ES001 was developed by O'Connor and several references may be consulted to supplement this development (1,2,3,4).

ES001 is meant to be used to either furnish insight into a particular phenomena, or as a predictive device for use in water quality planning.

Care should be taken at all times to consider all the assumptions underlying its formulation and by no means could it ever be construed to apply to any and every estuarine situation. With this in mind it is an excellent tool for the use of those interested in applying rational approaches to the problems of the deterioration of the environment.

THEORY

The Law of Conservation of Mass is the basis from which the ES001 model has been developed. Using this law and making certain assumptions, differential equations may be derived which describe the behavior of a mass in a one-dimensional estuary. Since the assumptions used to derive the equations state that the body of water have constant hydraulic and geometric characteristics, the estuary must be divided into segments in which parameters, such as cross-sectional area, depth, width, etc., are constant. Segments are also formed at points of waste input. Individual differential equations are then written for each segment and solved in terms of unknown coefficients. The segments are then joined in a mathematical fashion, by means of boundary conditions relating concentration and flux of mass at each face of the junctions between the segments, and the unknown coefficients are determined mathematically. Finally, the coefficients are placed in the original solutions which are then used to generate continuous values of the parameter under study throughout each section.

The following sections are designed to introduce the user to the theory upon which program ES001 is based, as well as the analytic forms in which this theory is applied in the program. The chapter is divided into two areas: a theoretical discussion and a section on the specific way in which the theory is applied in ES001.

The former includes a section on basic theory, a development of water quality equations for an estuary, sections on segmentation of the system and evaluation of boundary conditions and a sample problem. The latter is basically a review of the evaluation and presentation of boundary conditions as particular to program ES001.

In general, discussion will focus on the BOD-DO deficit system. However, the program is capable of modeling analogous systems of what might be called sequential reactions of two substances having first order kinetics. For instance ES001 can be used to model a nitrogen reaction with ammonia and nitrate analogous to BOD and D.O. deficit, respectively.

Basic Theory:

The distribution of a substance in a natural body of water is governed by the law of conservation of mass; that is, the rate of change with respect to time of a mass within a volume must be equal to the net influx of mass. For a one-dimensional estuary this may be expressed in a general way by:

$$\frac{\partial c}{\partial t} = \frac{E}{A}\frac{\partial}{\partial x}\left(A\frac{\partial c}{\partial x}\right) - \frac{Q}{A}\frac{\partial c}{\partial x} \pm \Sigma S \quad . . . . (1)$$

where

c = concentration of some mass (e.g., BOD or DO deficit) $[M/L^3]$*

t = time $[T]$

E = dispersion coefficient $[L^2/T]$ which incorporates diffusion effects caused by turbulence, velocity gradients, density currents as well as the mixing brought about by the tidal velocity. ES001 does not model the concentration of substances within the tidal cycle but rather looks at the distribution of the mass as it would behave over a longer time span while incorporating the effects of the tides in this coefficient.

A = cross-sectional area of the estuary $[L^2]$. O'Connor has developed equation 1 for variable areas, $A(x)$,[6]. These solutions are not suited for this general approach and therefore, cross-sectional areas are approximated by constant values in ES001.

x = distance along the estuary $[L]$.

Q = net advective flow of the estuary $[L^3/T]$. As implied in the definition of E, this does not include the tidal velocity. In many estuaries this is best described as the "fresh water flow."

$\pm\Sigma S$ = the various sources and sinks of the mass in question. For DO a sink of oxygen may be a deposit of sludge on the streambed which exerts a demand on the oxygen in the estuary while a source might be the oxygen which enters the water from the atmosphere.

---

*Letters in brackets designate units. M is mass, L is length and T is time.

Program ES001 generates solutions for equation 1 when it has constant coefficients (e.g., area is constant) and when it is at steady state ($\partial c/\partial t = 0$). Equation (1) may then be written as follows for a conservative substance like chlorides:

$$0 = E\frac{d^2\zeta}{d_x{}^2} - U\frac{d\zeta}{dx} + Y_c \quad \dots\dots\dots\dots\dots\dots\dots\dots \quad (2)$$

where

ζ = concentration of conservative substance [M/L^3]

U = Q/A = net advective velocity [L/T]

Y_c = sources and sinks of conservative substance [$M/(L^3T)$]

= 0 (no sources or sinks of conservative substance will be considered in ES001)

For a non-conservative substance such as BOD, equation 1 can be rewritten as:

$$0 = E\frac{d^2L}{d_x{}^2} - U\frac{dL}{dx} - RK\cdot L + Y_b \quad \dots\dots\dots\dots\dots \quad (3)$$

where

L = concentration of BOD [M/L^3].

U = Q/A = net advective velocity [L/T].

RK = removal coefficient of BOD [1/T]. This represents the rate at which BOD is removed from the system by a number of means (usually assimilation by the organisms in the estuary). As can be seen the assumption is made that removal is of first order reaction kinetics. This holds that the rate of reaction of a substance is directly proportional to the concentration of that substance at any time, t. This assumption is justified by the fact that first order kinetics have been found to be an excellent approximation to a large body of reactions in natural water systems and that the equations are more amenable to solution.

Y_b = additional sources and sinks of BOD [$M/(L^3T)$].

For DO deficit, equation 1 can be rewritten as:

$$0 = E\frac{d^2D}{d_x{}^2} - U\frac{dD}{dx} - AK\cdot D + Y_d \quad \dots\dots\dots\dots\dots \quad (4)$$

where

D = DO deficit $[M/L^3]$ = C_s - DO

C_s = saturation value of dissolved oxygen in water at a specific temperature and pressure $[M/L^3]$

DO = concentration of dissolved oxygen $[M/L^3]$

AK = reaeration coefficient [1/T]. The reaction rate represents the rate at which oxygen enters the body of water from the atmosphere. It is assumed to follow first order kinetics.

Y_d = additional sources and sinks of DO deficit $[M/(L^3T)]$

Equations (2), (3) and (4) can be integrated to yield:

$$\zeta = Ze^{(U/E)\cdot X} + J \tag{5}$$

$$L = Be^{S\cdot X} + Ce^{V\cdot X} + YB \tag{6}$$

$$D = Ge^{SD\cdot X} + He^{VD\cdot X} + YD \tag{7}$$

where Z, J, B, C, G and H are constants to be evaluated by consideration of the boundary conditions.

S and V Contain the parameters U, E, RK

SD and VD Contain the parameters U, E, AK

YB = The particular integral of the BOD source and sink terms

YD = The particular integral of the deficit source and sink terms and which always includes the BOD solution equation

Equations (5), (6) and (7) were developed on the assumption of constant values of the parameters U, E, RK, and AK. These equations are applicable to any section of an estuary where the assumption of constant parameter values is valid. The estuary must be segmented at locations where the value of a parameter changes, at locations of point waste inputs and at locations of changes in the sources or sinks of material. Figure 1 illustrates such segmentation. The appropriate equation

(Equation (5), (6) or (7), depending on the substance being studied) is then applied to each of the sections formed by the segmentation process. The unknown coefficients are evaluated by considering the boundary conditions at the junction points of adjacent segments. In effect, the boundary conditions at each junction of sections couples the several segments into a single system interrelated through the values of the appropriate unknown coefficients. Considering the BOD solution as an example, the sections formed by segmentation are coupled into a single system by evaluation of boundary conditions at the junction points between sections. The values of the unknown coefficients B and C in any section will reflect the phenomena occurring in all other sections of the system.

The total number of boundary conditions that must be evaluated in any system is equal to twice the number of segments in the system. There are three classes of boundary condition equations which can be specified. These are:

1. Infinity conditions:

   At locations remote from a waste source, the anticipated concentration attributable to the waste source should approach or be equal to zero. This can be stated mathematically by evaluating the appropriate equation, at positive or negative infinity.

2. Concentration equality:

   At the junction of two segments, the concentration calculated by the equation in either segment must be the same. Thus, the profile equation in the upstream segment must equal the profile equation in the downstream segment at the junction of the two segments.

3. Mass balance:

   It is possible to develop a material balance around an element at the junction of two sections. The quantity of the substance entering the element must equal the quantity of substance leaving the element.

Proper application of the three general classes of boundary conditions indicated above will generate sufficient equations to evaluate all unknown coefficients.

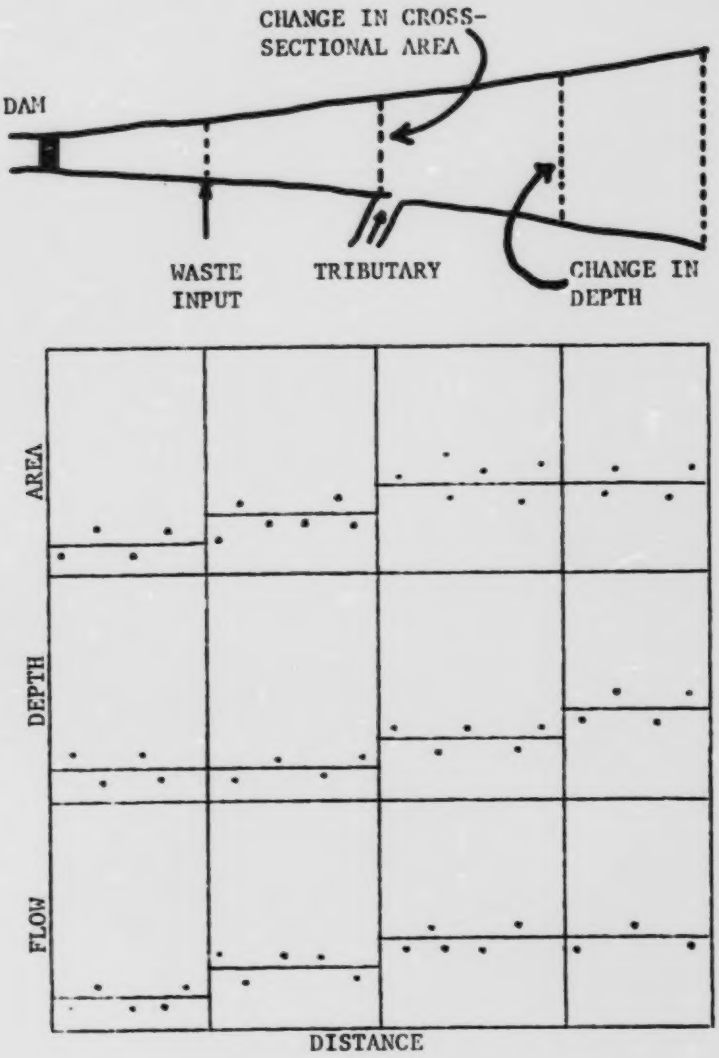


Figure 1. Illustration of segmentation.

At every junction of "n" segments (n must be greater than one), a single mass balance equation, and (n - 1) concentration equations can be specified. If one end of a section is not bounded by adjacent segments, the infinity boundary condition may be applied.

Evaluation of the boundary condition equations linking adjacent segments results in a series of simultaneous equations equal to the number of unknown coefficients. It is convenient to arrange these unknown equations into a matrix and employ standard techniques to evaluate each of the unknown coefficients. In order to facilitate setting up the matrix of unknown coefficients, a series of general boundary condition equations have been developed and tabulated in this report. The boundary equations are presented in terms of the elements of the matrix and the forcing functions. All specific boundary condition equations required to mathematize the study area have been included in the tabulation.

Development of Water Quality Equations for an Estuary:

The System

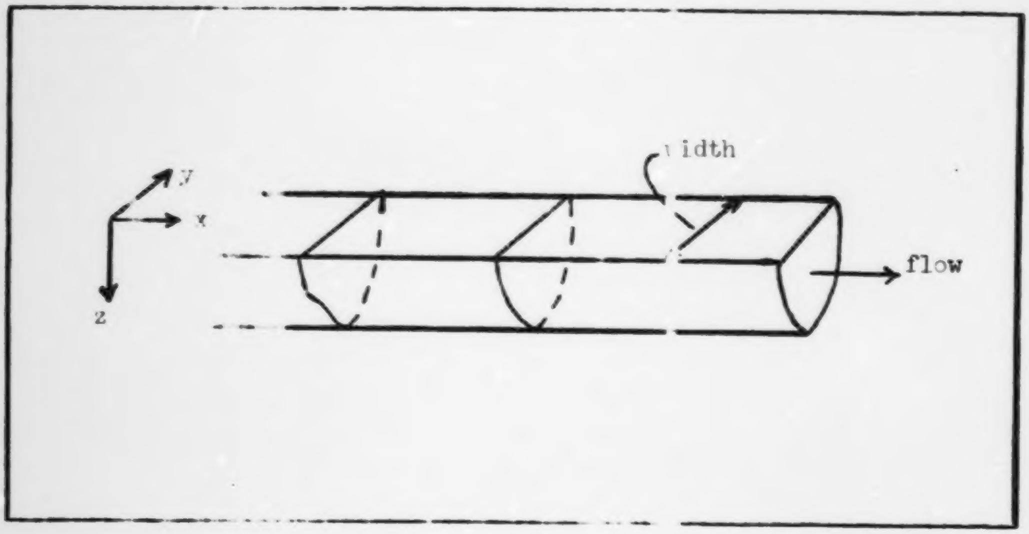


Figure 2. An Estuary.

For the purpose of analysis, the estuary is defined as that portion of the river which is under the influence of tidal action in which the dispersion factor is always significant. The advective term may or may not be significant depending on the rate of fresh-water flow and the cross-sectional area. As noted previously, the advective term is a net advection or fresh water flow and does not include the tidal velocity. Any mixing due to the tidal velocity is included in the bulk dispersion rate, E, which also takes into account the effects of turbulence, velocity gradients and density currents.

The particular system of ES001 is one dimensional in nature. This implies that the substance being modelled is assumed to be uniformly distributed in the Y and Z directions as illustrated in Figure 2.

Conservative Substance

A conservative substance is one which does not decay significantly when entering an estuary. As such, chlorides and some pesticides might be considered in this category.

The equations for evaluation of conservative substances can be developed by considering a differential element of an estuary as shown in Figure 3:

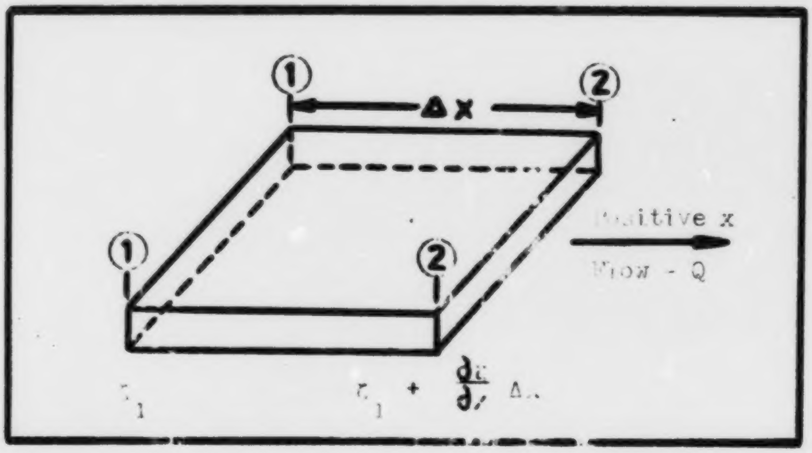


Figure 3. Differential Element of an Estuary (conservative substance).

A mass balance is obtained over the time interval Δt since the law of conservation of mass applies. The buildup in the element is evaluated from a consideration of the material entering and leaving through advective and dispersive transport.

In this particular case, no source and sinks are considered as it is assumed that the concentration of a conservative substance is only affected by advective and dispersive forces.

Mass Balance:

entering element:

advection: $Q_1 \zeta_1$

dispersion: $-E_1 A_1 \frac{\partial \zeta_1}{\partial x}$

leaving element:

advection: $Q_2 (\zeta_1 + \frac{\partial \zeta}{\partial x} \Delta X)$

dispersion: $-E_2 A_2 \frac{\partial}{\partial x} (\zeta_1 + \frac{\partial \zeta}{\partial x} \Delta X)$

Equation:

$$\text{VOL} \frac{\partial \zeta}{\partial t} = Q_1 \zeta_1 - E_1 A_1 \frac{\partial \zeta_1}{\partial x} - Q_2 (\zeta_1 + \frac{\partial \zeta}{\partial x} \Delta X)$$

$$+ E_2 A_2 \frac{\partial}{\partial x} (\zeta_1 + \frac{\partial \zeta}{\partial x} \Delta X)$$

VOL = volume

assume

$$Q = Q_1 = Q_2, \quad E = E_1 = E_2, \quad A = A_1 = A_2$$

and divide by elemental volume, $A \Delta X$, and simplify

$$\frac{\partial \zeta}{\partial t} = -\frac{Q}{A} \frac{\partial \zeta}{\partial x} + E \frac{\partial}{\partial x} (\frac{\partial \zeta}{\partial x})$$

at steady state $(\partial\zeta/\partial t = 0)$

$$0 = -\frac{Q}{A}\frac{\partial\zeta}{\partial x} + E\frac{\partial^2\zeta}{\partial x^2}$$

$$Q/A = U$$

$$0 = F\frac{\partial^2\zeta}{\partial x} - U\frac{\partial\zeta}{\partial x} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (8)$$

as can be seen we have derived equation (2).

Solution:

Equation (8) is a linear homogeneous second order differential equation with constant coefficients whose general solution is

$$\zeta = Ze^{(U/E)\cdot X} + J \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (9)$$

where

Z and J are unknown coefficients to be determined via boundary conditions.

Biochemical Oxygen Demand (BOD)

BOD has historically been one of the major measures of the pollutional strength of a water. It is usually defined as the amount of oxygen required by organisms while stabilizing decomposable organic matter under aerobic conditions. It is analyzed quantitatively by placing a dose of the waste into a bottle whose contents simulate the natural aerobic water environment as closely as possible; that is, water which is saturated with oxygen, has the necessary nutrients, has proper osmotic and temperature conditions, etc. The depletion in oxygen is then recorded over intervals of time and quantitative values of BOD are calculated. Figure 4 shows the typical plot of these BOD values for a water containing organic matter as well as some nitrogen.

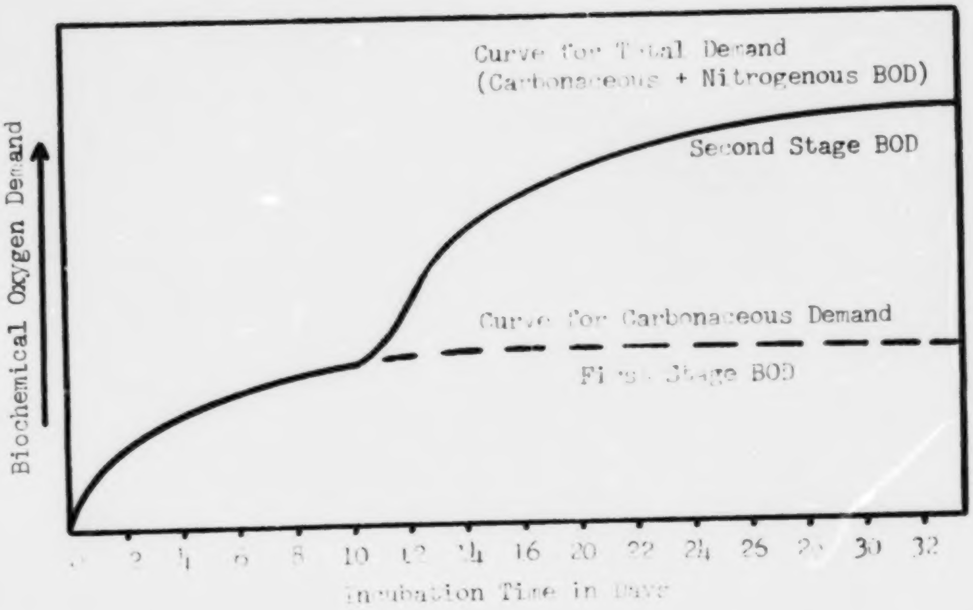


Figure 4. BOD Curve for Typical Municipal Sewage.

As can be seen the reaction essentially takes place in two stages: a carbonaceous stage (CBOD) during which about 70 to 80 percent of the organic carbon is oxidized and a nitrogenous stage (NBOD) during which biochemical oxidation of ammonia nitrogen and organic nitrogen is predominant. Since these reactions take place in a bottle and because the actual reaction as it occurs in the stream is not fully understood, there is some speculation as to the applicability of such test results to actual water systems. At the present time, it is believed that highly treated effluents will exert both first and second stage BOD simultaneously. Untreated effluents will immediately exert a CBOD while the NBOD will exert itself after a lag period.[7] Another factor which is recognized is that at low levels of DO (approximately <1.5 mg/l) nitrification practically ceases.[8] The nitrogenous component can often be significant[9] and the misunderstanding concerning its behavior is such that the user must take care in modelling this parameter. The ES001 program is capable of modelling NBOD as well as the conventional CBOD merely by inputting the appropriate loadings and reaction rates for each component. The DO deficit responses from each may be added to determine the total deficit attributable to them. All theoretical development will be done in terms of "BOD" which can signify either component.

ES001 is also capable of analyzing either ultimate BOD or 5-day BOD. A value of, F,

$$F = \frac{\text{ultimate BOD}}{\text{5-day BOD}}$$

can be inputted for individual sections of the estuary when 5-day values are being used. (When the ultimate values are used F is inputted as 1). This option can be particularly useful in the verification of some models as 5-day BOD's are much easier to obtain than ultimates. Of course, use of this option depends on the types of waste involved and whether the assumption of a uniform F for a section is realistic. ES001 will output 5-day BOD and the actual deficit in the stream when the option is exercised.

The equations for evaluation of BOD in an estuary can be developed by considering a differential element of an estuary as shown in Figure 5.

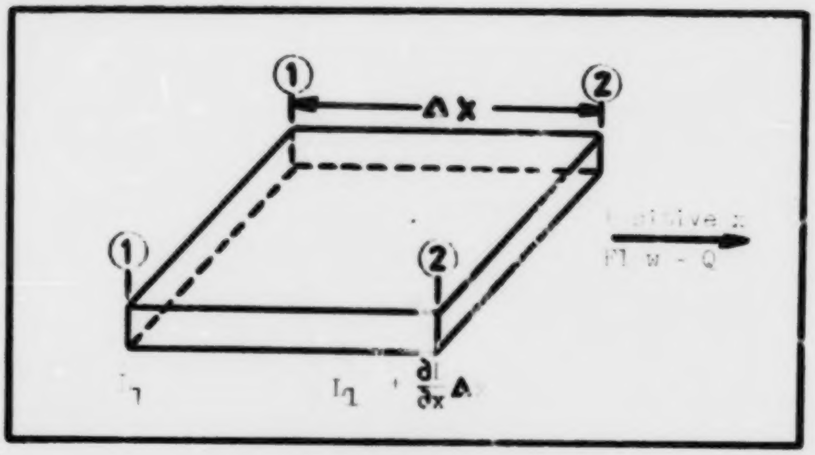


Figure 5. Differential Element of an Estuary (BOD).

First, a simple case involving no waste loads will be examined to illustrate the derivation of equation 3. Then a second more complex case involving uniform loads will be derived. It is this second case which is contained in ES001.

1) Case 1 - BOD with no waste loads

A mass balance is obtained, over the time interval Δt, since the law of conservation of mass applies. The buildup in the element is evaluated from a consideration of the material entering and leaving through advective and dispersive transport. The contribution of sources and sinks of material within the element are also included in the inventory.

Mass Balance: no waste load

entering element:

advection: $Q_1 \; L_1$

dispersion: $- E_1 \; A_1 \; \frac{\partial L_1}{\partial X}$

leaving element:

advection: $Q_2 \; (L_1 + \frac{\partial L}{\partial X} \; \Delta X)$

dispersion: $- E_2 \; A_2 \; \frac{\partial}{\partial x} \; (L_1 + \frac{\partial L}{\partial X} \; \Delta X)$

sink:

BOD removal: $VOL \cdot L \cdot RK$

Equation:

$$VOL \frac{\partial L}{\partial t} = Q_1 \; L_1 - E_1 \; A_1 \; \frac{\partial L_1}{\partial X} - Q_2 \; (L_1 + \frac{\partial L}{\partial x} \; \Delta X)$$

$$+ E_2 \; A_2 \; \frac{\partial}{\partial X} \; (L_1 + \frac{\partial L}{\partial X} \; \Delta X) - VOL \cdot L \; RK$$

assume

$$Q = Q_1 = Q_2, \quad E = E_1 = E_2, \quad A = A_1 = A_2$$

and divide by elemental volume, AΔX, and simplify

$$\frac{\partial L}{\partial t} = - \frac{Q}{A} \; \frac{\partial L}{\partial X} + E \; \frac{\partial}{\partial X} \; (\frac{\partial L}{\partial X}) - L \cdot RK$$

at steady state ($\partial L/\partial t = 0$):

$$0 = - \frac{Q}{A} \; \frac{dL}{dX} + E \; \frac{d^2L}{dX^2} - L \cdot RK$$

$$Q/A = U$$

$$0 = E \frac{d^2L}{dx^2} - U \frac{dL}{dx} - L \cdot RK \quad \text{. . . . .} \quad (10)$$

As can be seen, equation (3) has been derived in a slightly more sophisticated form.

Solution:

Equation (10) is a linear homogeneous second order differential equation with constant coefficients whose general solution is

$$L = Be^{Sx} + Ce^{Vx} \quad \text{. . . . .} \quad (11)$$

where

B and C are unknown coefficients to be determined via boundary conditions and

$$S,V = \frac{U}{2E} \left[1 \pm \left[1 + \frac{4E \cdot RK}{U^2}\right]^{1/2}\right] \quad \text{. . . . .} \quad (11a)$$

2) Case 2 - Now the effect of a uniform discharge of waste on the BOD will be examined.

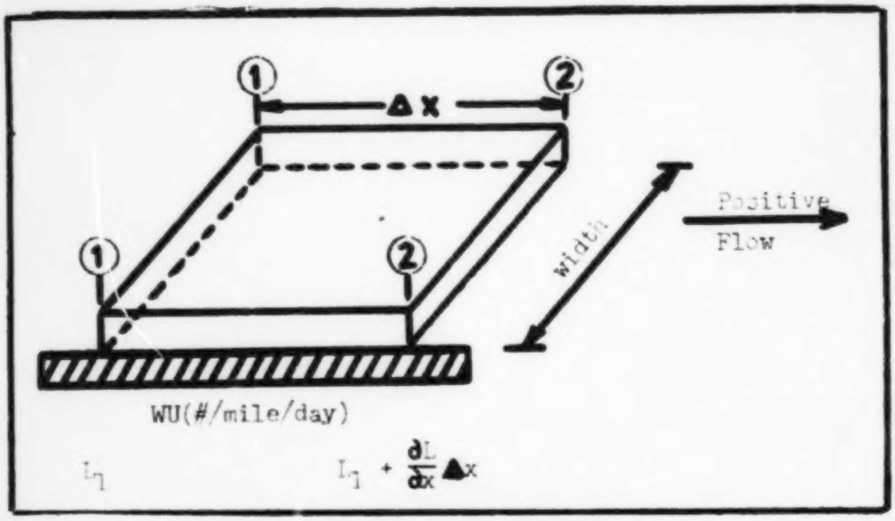


Figure 6. BOD with Uniform Load.

Mass Balance:

entering element:

$$Q_1 L_1 - E_1 A_1 \frac{\partial L_1}{\partial X}$$

leaving element:

$$Q_2 \left(L_1 + \frac{\partial L}{\partial X} \Delta X\right) - E_2 A_2 \frac{\partial}{\partial X} \left(L_1 + \frac{\partial L}{\partial X} \Delta X\right)$$

source: uniform load = $WU \cdot \Delta X$

sink: $VOL \cdot L \cdot RK$

Equation:

$$VOL \frac{\partial L}{\partial t} = Q_1 L_1 - E_1 A_1 \frac{\partial L_1}{\partial X} - Q_2 \left(L_1 + \frac{\partial L}{\partial X} \Delta X\right)$$

$$+ E_2 A_2 \frac{\partial}{\partial X} \left(L_1 + \frac{\partial L}{\partial X} \Delta X\right) - VOL \cdot L \cdot RK + WU \cdot \Delta X$$

assume

$$E = E_1 = E_2, \quad A = A_1 = A_2, \quad Q = Q_1 = Q_2 \quad \text{and}$$

divide by elemental volume, $A\Delta X$, set equation at steady state and simplify

$$0 = -\frac{Q}{A} \frac{dL}{dX} + E \frac{d^2L}{dx^2} - L \cdot RK + \frac{WU}{A}$$

$Q/A =$

$$\frac{WU}{A} = -E \frac{d^2L}{dx^2} + U \frac{dL}{dx} + L \cdot RK \quad \ldots\ldots\ldots\ldots\ldots \quad (12)$$

Solution:

Equation (12) is a linear nonhomogeneous second order differential equation with constant coefficients whose general solution is the same as in Case [illegible] (equation (11))

$$L_g = Be^{SX} + Ce^{VX}$$

The particular solution is

$$L_p = WU/(A \cdot RK) \quad \text{. . . . . (12a)}$$

The total solution is therefore

$$L = Be^{SX} + Ce^{VX} + \frac{WU}{A \cdot RK} \quad \text{. . . . . (13)}$$

Dissolved Oxygen Deficit

Oxygen is soluble in water to a limited extent, being directly related to the partial pressure of oxygen in the atmosphere. The saturation value of oxygen is primarily a function of salinity and temperature. The difference between the saturation concentration and the actual concentrations of oxygen is defined as the dissolved oxygen deficit.

Dissolved oxygen in natural waters can be employed to satisfy the oxygen demand of pollutants, as measured by the BOD. In the development of the equations for BOD, the rate of BOD removal was defined by the parameter "RK". In a natural body of water, the rate of BOD removal does not necessarily equal the rate at which oxygen is utilized to satisfy the BOD. If organic matter, having a BOD, is removed by sedimentation, stripping, or other physical means, oxygen is not consumed in this removal of BOD. Since these phenomena can occur, it is necessary to employ a parameter in the dissolved oxygen material balance which indicates the rate at which oxygen is utilized to satisfy BOD. This parameter is defined as "DK". The parameter "DK" can be equal to, or less than the parameter "RK".

The rate at which oxygen is utilized is equal to the product of the parameter "DK" and the ultimate BOD present. As the oxygen concentration is reduced below the saturation level, the deficit is made up by atmospheric oxygen going into solution. This phenomenon is termed atmospheric reaeration. It occurs at a rate equal to the product of the dissolved oxygen deficit and the reaeration coefficient, "AK".

The differential equation defining dissolved oxygen deficit may be developed by constructing a material balance around a differential element and including the various sources and sinks of dissolved oxygen within the volume.

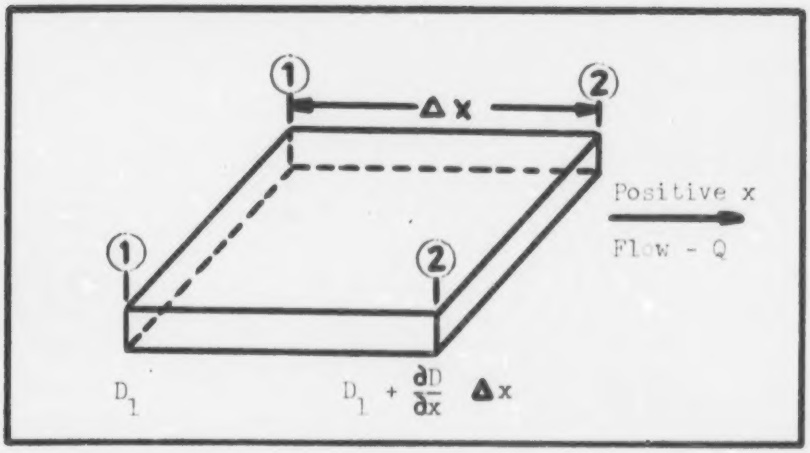


Figure 7. Differential Element of an Estuary (DO deficit).

As in the development of the BOD equations, a simple case involving no loadings will be derived first. It will be followed by the derivation of the more sophisticated case which is used in program ES001. This second case includes benthal demands and the effects of photosynthesis and respiration of algae.

1) Case 1 - DO deficit with no waste load

The DO deficit is constructed in a similar fashion to the BOD. A balance is made between the amounts entering, leaving, and any sources or sinks within the element.

Mass Balance:

entering element:

$$Q_1 D_1 - E_1 A_1 \frac{\partial D_1}{\partial x}$$

leaving element:

$$Q_2 (D_1 + \frac{\partial D}{\partial X} \Delta X) - E_2 A_2 \frac{\partial}{\partial X} (D_1 + \frac{\partial D}{\partial X} \Delta X)$$

source: deoxygenation due to BOD $= DK \cdot L_x \cdot VOL$

sink: reaeration $= AK \cdot D \cdot VOL$

Equation:

$$VOL \frac{\partial D}{\partial t} = Q_1 D_1 - E_1 A_1 \frac{\partial D_1}{\partial X} - Q_2 (D_1 + \frac{\partial D}{\partial X} \Delta X)$$

$$+ E_2 A_2 \frac{\partial}{\partial X} (D_1 + \frac{\partial D}{\partial X} \Delta X) + DK \cdot L_x \cdot VOL - AK \cdot D \cdot VOL$$

$$E = E_1 = E_2, \quad A = A_1 = A_2, \quad Q = Q_1 = Q_2 \text{ and}$$

divide by elemental volume, $A\Delta X$, set equation at steady state and simplify

$$DK \cdot L_x = - E \frac{d^2 D}{dx^2} + U \frac{dD}{dx} + AK \cdot D \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \quad (14)$$

The quantity "Lx" is the ultimate BOD concentration at a point X in the estuary and is obtained from the BOD solution.

$$L_x = [Be^{SX} + Ce^{VX} \qquad \cdot F$$

Solution:

Equation 14 is a linear nonhomogeneous second order differential equation with constant coefficients whose general solution is:

$$D_g = Ge^{SDX} + He^{VDX} \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \quad (15)$$

where G and H are unknown coefficients to be determined via boundary conditions.

$$SD, VD = \frac{U}{2E} [1 \pm [1 + \frac{4AK \cdot E}{U^2}]^{1/2}] \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \quad (15a)$$

The particular solution is

$$D_p = \frac{DK \cdot F}{AK-RK} \text{ (BOD SOLUTION)} \quad . . . . (16)$$

The BOD SOLUTION is equation (11), therefore, the complete solution is

$$D = Ge^{SDX} + He^{VDX} + \frac{DK \cdot F}{AK-RK} \text{ (BOD SOLUTION)} \quad . . . (17)$$

2) Case 2 - DO deficit with benthal demand and algal effects

The complete dissolved oxygen deficit scheme of an estuary as presented in this model includes several sources and sinks: uniform waste load, algae photosynthesis and respiration, and benthal demands.

Algae can influence the dissolved oxygen deficit profile. A daily cycle of algae respiration can be approximated by a constant respiration rate, "RS", and an average daily photosynthesis rate of oxygen production, "P". The units of "RS" and "P" are the mass of oxygen used or produced per day per unit volume.

Benthal demands result from sludge deposits on the streambed. A bottom oxygen demand, BD, can be formulated, the units of which are mass of demand per area of bottom per day.

Figure 8 will be referred to when deriving the complete DO deficit solution used by this model.

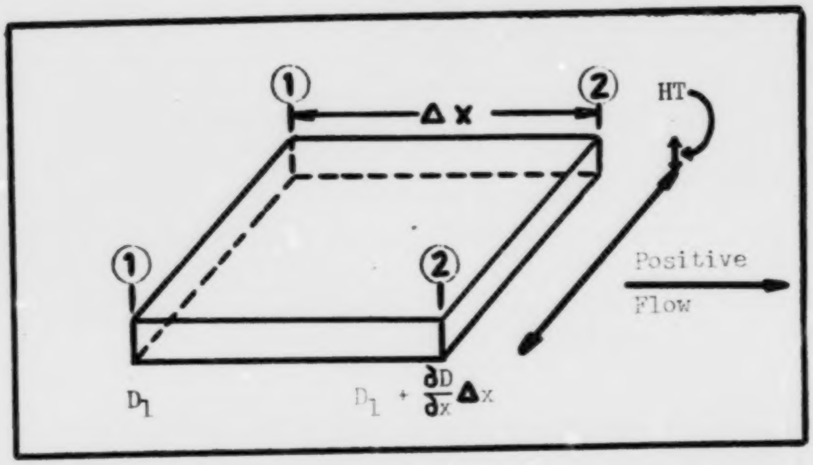


Figure 8. Differential Element of Estuary for DO deficit Approach.

Mass Balance:

entering element:

$$Q_1 D_1 - E_1 A_1 \frac{\partial D_1}{\partial X}$$

leaving element:

$$Q_2 (D_1 + \frac{\partial D_1}{\partial X} \Delta X) - E_2 A_2 \frac{\partial}{\partial X} (D_1 + \frac{\partial D}{\partial X} \Delta X)$$

sources: deoxygenation due to BOD = $DK \cdot L_x \cdot VOL$

respiration of algae: $RS \cdot VOL$

benthal demand: $BD \cdot \Delta X \cdot WI$

sinks: reaeration = $AK \cdot D \cdot VOL$

photosynthesis of algae = $P \cdot VOL$

Equation:

$$VOL \frac{\partial D}{\partial t} = Q_1 D_1 - E_1 A_1 \frac{\partial D_1}{\partial X} - Q_2 (D_1 + \frac{\partial D}{\partial X} \Delta X)$$

$$+ E_2 A_2 \frac{\partial}{\partial X} (D_1 + \frac{\partial D}{\partial X} \Delta X) + DK \cdot L_x \cdot VOL + RS \cdot VOL$$

$$+ BD \cdot \Delta X \cdot WI - AK \cdot D \cdot VOL - P \cdot VOL$$

$E = E_1 = E_2$, $A = A_1 = A_2$, $Q = Q_1 = Q_2$ and

divide by elemental volume, $A\Delta X$, set equation at steady state and simplify.

$$DK \cdot L_x + (RS-P) + \frac{BD \cdot \Delta X \cdot WI}{A\Delta X} = - E \frac{\partial^2 D}{\partial X^2} + U \frac{\partial D}{\partial X} + AK \cdot D$$

note: $\frac{WI \cdot \Delta X}{A \cdot \Delta X} = \frac{1}{HT}$ (where HT is water depth)

Therefore, the total equation is

$$DK \cdot L_x \qquad + (RS-P) + \frac{BD}{HT} = - E \frac{\partial^2 D}{\partial X^2} + U \frac{\partial D}{\partial X} + AK \cdot D \; . \; . \; . \; . \qquad (18)$$

Where

$$L_x = (Be^{SX} + Ce^{VX} + \frac{WU}{RK \cdot A}) F$$

Solution:

Equation 18 differs from equation 14 in its nonhomogeneous part so the general solutions must be the same:

$$D_g = Ge^{SDX} + He^{VDX}$$

The particular solutions for each source and sink are:

$$\text{deoxygenation:} \quad \frac{DK \cdot F}{AK-RK} \; [Be^{SX} + Ce^{VX}] + \frac{F \cdot DK \cdot WU}{AK \cdot RK \cdot A}$$

$$\text{respiration and photosynthesis:} \quad \frac{RS-P}{AK} \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \qquad (20)$$

$$\text{benthal demand:} \quad \frac{BD}{AK \cdot HT} \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \qquad (21)$$

Letting RA = RS-P and calling it the "net algal effect", the complete solution is therefore

$$D = Ge^{SD \cdot X} + He^{VD \cdot X} + \frac{BD}{HT \cdot AK} + \frac{RA}{AK}$$

$$+ \frac{F \cdot DK \cdot WU}{AK \cdot RK \cdot A} + \frac{DK}{AK-RK} \; [Be^{SX} + Ce^{VX}] \; F \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \qquad (22)$$

Table 1 presents a summary of the applicable equations for constant cross-sectional area estuaries for several loading conditions and phenomena which normally influence water quality in an estuary.

TABLE 1

SUMMARY OF EQUATIONS FOR STEADY - STATE SECTION ESTUARY ANALYSIS

Application	BOD	Dissolved Oxygen Deficit
Upstream or downstream load	$L = Be^{SX} + Ce^{VX}$	$D = Ge^{SDX} + He^{VDX} + [\frac{DK \cdot F}{AK-RK}]\ [Be^{SX} + Ce^{VX}]$
Upstream or downstream load with bottom oxygen demand and algae	$L = Be^{SX} + Ce^{VX}$	$D = Ge^{SDX} + He^{VDX} + \frac{BD}{HT \cdot AK} + \frac{RA}{AK} + [\frac{DK \cdot F}{AK-RK}]\ [Be^{SX} + Ce^{VX}]$
Uniform BOD load WU (#/day/mi) in section	$L = Be^{SX} + Ce^{VX} + \frac{WU}{A \cdot RK}$	$D = Ge^{SDX} + He^{VDX} + [\frac{DK \cdot F}{AK-RK}]\ [Be^{SX} + Ce^{VX}] + \frac{DK \cdot WU \cdot F}{AK \cdot RK \cdot A}$
Upstream or downstream load with uniform BOD load, algae & bottom deposit in the section	$L = Be^{SX} + Ce^{VX} + \frac{WU}{A \cdot RK}$	$D = Ge^{SDX} + He^{VDX} + \frac{BD}{HT \cdot AK} + \frac{RA}{AK} + \frac{DK \cdot WU \cdot F}{AK \cdot RK \cdot A} + \frac{DK \cdot F}{AK-RK}\ [Be^{SX} + Ce^{VX}]$

Other Applications:

Program ES001 was developed to model the Biochemical Oxygen Demand (BOD), Dissolved Oxygen (DO) Deficit system which is presently the most common measure of estuarine pollution. For this reason this documentation focuses on this system. However, by analogy ES001 may also be employed to model the distribution of other important substances. This is true because the BOD-DO deficit system is representative of a general class of reactions.

The BOD-DO deficit reaction might be generally described as a "coupled system of two non-conservative substances which react with first order kinetics." A schematic of this reaction is shown in Figure 9:

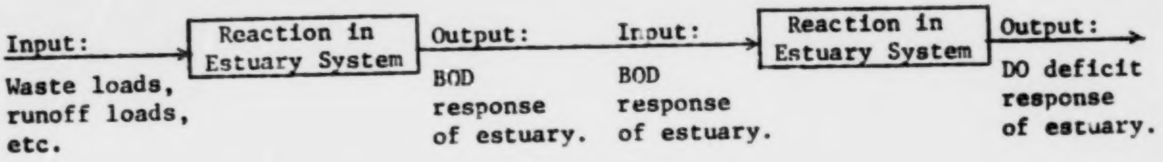

Input:
Waste loads, runoff loads, etc.
Reaction in Estuary System
Output:
BOD response of estuary.
Input:
BOD response of estuary.
Reaction in Estuary System
Output:
DO deficit response of estuary.

Figure 9. BOD-DO Deficit System.

This oversimplified representation shows that in this particular system, the basic input of waste load forces an output of BOD response from the estuary system. This output in turn forms an input to the system which forces a DO deficit response. Other systems might be represented in analogous fashion as shown in Figure 10 if the assumptions of the kinetics is justifiable.

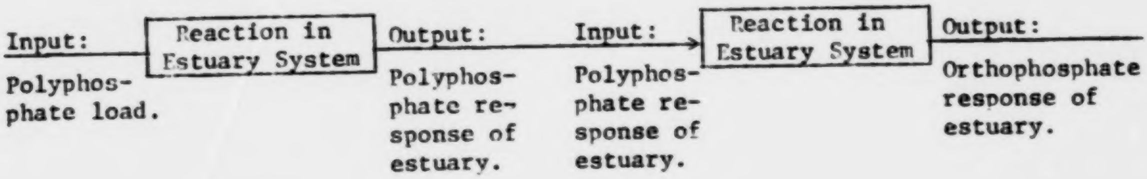

Input:
Polyphosphate load.
Reaction in Estuary System
Output:
Polyphosphate response of estuary.
Input:
Polyphosphate response of estuary.
Reaction in Estuary System
Output:
Orthophosphate response of estuary.

Figure 10. Polyphosphate-Orthophosphate System.

Program ES001 is capable of modelling this type of reaction as well as some subsets of it. A list of the options available using ES001 are shown in Table 2.

Segmentation of Estuary Systems:

Equations have been developed in the preceding sections for estuaries having constant values of the parameters. Specifically, it was assumed in each of the developments that the parameters AK, RK, DK, E, A, and Q were constant. The solutions thus derived are valid for any length of estuary in which the values of the individual parameters do not change. At the location of changes in the value of individual parameters, it is necessary to segment the system and apply the appropriate general solution to each section.

Additional segments are required for any of the following conditions which would cause a change in a parameter.

1. The magnitude of the dispersion phenomena may be substantially reduced upstream of the point of maximum salt intrusion.
2. Fresh water flow may enter a system thus altering the advective velocity and the dilution factor.
3. The cross-sectional area of the body of water may change, thus altering the fresh water velocity and system volume.
4. Segmentation is required at the confluence or two or more bodies of water.

For the specific BOD-DO deficit system:

5. Oxidation of nitrogenous compounds may not occur at the point of waste introduction, or may require substantial oxidation of carbon sources and the presence of high dissolved oxygen concentrations. The system is segmented at the point of the beginning of nitrogenous oxidation.
6. The rate of BOD removal RK may include the effects of settling, stripping, etc., near the location of a raw discharge. In this instance, the rate of BOD removal may be greater than the rate of oxygen utilization, DK. Following the completion of settling or stripping, the rate of BOD removal may be equal to the rate of oxygen utilization. Segmentation is necessary when the rate of BOD removal RK changes value.

TABLE 2

System	Reaction Discussed in This Documentation	Other Applications
Non-conservative substances (coupled system, first order reactive)	BOD-DO deficit (p. 15-26)	Carbonaceous BOD-DO deficit* Nitrogenous BOD-DO deficit* Polyphosphate-orthophosphate, etc.
Non-conservative substances (single system, first order reactive)	BOD (p. 15-21)	Coliform bacteria, etc.
Conservative substances (single system)	Conservative substances (p. 13-14)	Chlorides Total Dissolved Solids Pesticides (if their decay rate is slow enough), etc.

---

*ES001 has included the option to calculate each individual components of BOD and then add their DO deficit to give a total deficit.

7. Segmentation is required at the beginning and end of uniform BOD loads, bottom oxygen demands or algal demands.

8. Toxic effects may begin or end resulting in changes in the rate of BOD or nitrogenous oxidation. Segmentation is required when changes in rate occur.

9. The depth and/or velocity in an estuary may change, thus altering the value of AK, the reaeration coefficient.

10. Systems are segmented at the beginning and end of sections where oxygen is produced by algae or rooted aquatic plants.

In addition, estuary systems must be sectioned at the location of all waste inputs which are to be considered. There are two methods available to evaluate the influence of multiple waste inputs. It is possible to construct a model which includes all waste inputs thus requiring system segmentation at each input location.

A second method of evaluating multiple waste loads employs the superposition technique which is valid for the linear equations describing BOD and dissolved oxygen profiles. In the superposition method, the effect of individual waste loads can be evaluated separately. The cumulative effect of loads can be determined by adding the individual BOD and dissolved oxygen deficit profiles.

Evaluation of Boundary Conditions:

General equations have been presented describing conservative substance, BOD and dissolved oxygen deficit profiles in estuary systems. The appropriate equations are applicable to each section of the segmented system. Since the differential equations which were developed in the preceding sections are second degree, their solutions contain two arbitrary constants. The unknown coefficients Z and J in the conservative equation, B and C in the BOD equation and G and H in the deficit equation are evaluated by means of boundary conditions. Evaluations of these conditions also provide a method of linking adjacent system sections which were segmented in accordance with the criteria discussed in the previous section. Since the solution in each segment contains two unknown coefficients, the total number of boundary conditions that must be evaluated in any system is therefore equal to twice the number of segments in the system.

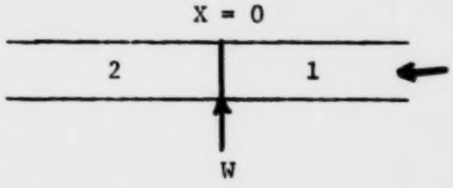
There are three classes of boundary conditions which are appropriate:

1. Infinity conditions - At locations remote from a waste source, the anticipated conservative substance, BOD and dissolved oxygen deficit attributable to the waste source should approach or be equal to zero. This can be stated mathematically by evaluating the BOD or dissolved oxygen deficit equation, at positive or negative infinity.

2. Concentration equality - At the junction of two segments, the concentration calculated by the equation in either segment must be the same. Thus, the profile equation in the upstream segment must equal the profile equation in the downstream segment at the location of the junction.

3. Mass Balance - It is possible to develop a material balance around an element at the junction of two sections. The quantity of substance entering the element must equal the quantity of BOD leaving the element.

Proper application of the three general classes of boundary conditions will generate sufficient equations to evaluate all unknown coefficients. In the following pages, examples of evaluation of boundary conditions will be presented for estuary systems.



In order to illustrate the specific application of boundary conditions, a simplified estuary will be considered with a single waste source of BOD located at x = 0. The estuary will be considered infinitely long in both directions and having constant cross-sectional area, flow and parameters AK, RK, DK and E. The estuary must be divided into two segments at the point of waste input. Section one extends upstream from the point of waste input and section two extends downstream from the point of waste discharge. The BOD equations applicable to each section are:

BOD: Section 1 (Upstream)

$$L_1 = B_1 e^{S_1 X} + C_1 e^{V_1 X} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (23)$$

BOD: Section 2 (Downstream)

$$L_2 = B_2 e^{S_2 X} + C_2 e^{V_2 X} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (24)$$

Note that each unknown coefficient and roots S and V are subscripted by section. In like manner the area, flow, reaction and dispersion are subscripted by section. The sectioning procedure results in four (4) unknown coefficients, thus requiring four (4) boundary conditions. The boundary conditions applicable to the example system are:

Boundary Condition	Location and Description	
#1	@ $X = -\infty$	$L_1 = 0$
#2	@ $X = +\infty$	$L_2 = 0$
#3	@ $X = 0$	$L_1 = L_2$
#4	@ $X = 0$	mass balance

Boundary Condition #1

At large distances upstream from the waste source, the BOD present in the estuary due to the waste discharges becomes small. Therefore, as X approaches negative infinity (- ∞), the BOD in section one approaches zero.

$$L_1 \longrightarrow 0 \text{ as } X \longrightarrow -\infty$$

$$L_1 = 0 \text{ at } X = -\infty$$

$$L_1 = B_1 e^{S_1 X} + C_1 e^{V_1 X}$$

S_1 = positive number

V_1 = negative number

$$e^{+S_1(-\infty)} = \frac{1}{e^{+S_1(+\infty)}} \longrightarrow 0$$

$$e^{-V_1(-\infty)} = e^{V_1(+\infty)} \longrightarrow \text{large number}$$

If $L_1 = 0$ at $X = -\infty$ $\quad C_1$ must $= 0$

Equation (23) for section one reduces to:

$$L_1 = B_1 e^{S_1 X}$$

Boundary Condition #2

Using the same reasoning:

$$L_2 \longrightarrow 0 \text{ as } X \longrightarrow +\infty$$

$$L_2 = 0 \text{ at } X = +\infty$$

$$L_2 = B_2 e^{S_2 X} + C_2 e^{V_2 X}$$

$$e^{+S_2(+\infty)} = e^{S_2(\infty)} \longrightarrow \text{large number}$$

$$e^{-V_2(+\infty)} = \frac{1}{e^{V_2(\infty)}} \longrightarrow 0$$

$$B_2 = 0$$

Equation (24) for section two reduces to:

$$L_2 = C_2 e^{V_2 X}$$

Boundary Condition #3

At X = 0, the BOD concentration calculated by the equation for the BOD profile in section one must equal the concentration as calculated using the BOD profile equation in section two.

$$L_1 = L_2 \text{ at } X = 0$$

$$B_1 e^{S_1 X} = C_2 e^{V_2 X}$$

$$@\ X = 0;$$

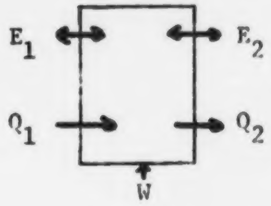
$$e^{S_1X} = e^{V_2X} = 1$$

and

$$B_1 = C_2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (25)$$

Boundary Condition #4

Considering a cross-sectional plane at X = 0, a flux or material balance can be obtained.



W = point source

(- X) X = 0

Material entering the section:

$$Q_1L_1 - E_1A_1 \frac{dL_1}{dX} + W$$

Material leaving the section:

$$Q_2L_2 - E_2A_2 \frac{dL_2}{dX}$$

Balance:

$$Q_1L_1 - E_1A_1 \frac{dL_1}{dX} + W = Q_2L_2 - E_2A_2 \frac{dL_2}{dX}$$

or

$$Q_2L_2 - Q_1L_1 + E_1A_1 \frac{dL_1}{dX} - E_2A_2 \frac{dL_2}{dX} - W = 0 \quad \ldots \quad (26)$$

Since

$$Q_2 = Q_1 \text{ and } L_2 = L_1 \text{ at } X = 0$$

$$E_1A_1 \frac{dL_1}{dX} - E_2A_2 \frac{dL_2}{dX} = W \quad . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \quad (27)$$

Evaluating

$$\frac{dL_1}{dX} \text{ and } \frac{dL_2}{dX} \text{ at } X = 0:$$

$\frac{dL_1}{dX}$:

$$\frac{dL_1}{dX} = B_1S_1e^{S_1X}$$

$$@ X = 0; \quad \frac{dL_1}{dX} = \underline{\underline{B_1S_1}}$$

$\frac{dL_2}{dX}$:

$$\frac{dL_2}{dX} = C_2V_2e^{V_2X}$$

$$@ X = 0; \quad \frac{dL_2}{dX} = \underline{\underline{C_2V_2}}$$

Substituting into Equation (27)

$$E_1A_1B_1S_1 - E_2A_2C_2V_2 = W$$

Grouping terms:

$$C_2\,(-E_2A_2V_2) + B_1\,(E_1A_1S_1) = W \quad . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \quad (28)$$

Summary of Boundary Equations

$$B_1 - C_2 = 0 \text{ (concentration equality)} \quad . \; . \; . \; . \; . \; . \; . \quad (29)$$

$$B_1\,(E_1A_1S_1) + C_2\,(-E_2A_2V_2) = W \text{ (mass balance)}. \; . \; . \; . \quad (30)$$

assuming

$$E_1 = E_2 = E \text{ and } A_1 = A_2 = A$$

There are two boundary condition equations and two unknown coefficients remaining, thus it is possible to solve for the unknown coefficients.

Combining equations:

$$B_1 \ (S_1 EA) \ - \ B_1 \ (V_2 EA) \ = W \tag{30A}$$

$$B_1 = \frac{W}{EA \ (S_1 - V_2)}$$

$$C_2 = \frac{W}{EA \ (S_1 - V_2)}$$

and

$$L_1 = \frac{W}{EA \ (S_1 - V_2)} \ e^{S_1 X} \tag{31}$$

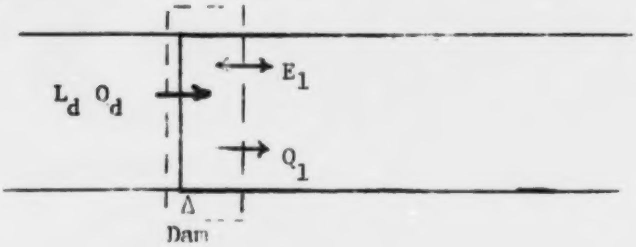
$$L_2 = \frac{W}{EA \ (S_1 - V_2)} \ e^{V_2 X} \tag{32}$$

The techniques for evaluating boundary conditions illustrated in the previous example can also be employed for evaluating the unknown coefficients in the dissolved oxygen deficit or conservative substance.

Additional Boundary Conditions:

There are two (2) special case boundary conditions which will be illustrated. These are for a system bounded by a dam, and an estuary entering a body of water with a fixed concentration.

Mass Balance at a Dam



Material entering the element:

$$QD \cdot LD$$

Material leaving the element:

$$Q_1 \cdot L_1 - E_1A_1 \frac{dL_1}{dX}$$

Balance: (in = out)

$$QD \cdot LD = Q_1L_1 - E_1A_1 \frac{dL_1}{dX} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (33)$$

The boundary condition can then be fully evaluated by substituting the appropriate equations for L_1 and $\frac{dL_1}{dX}$ into Equation (33).

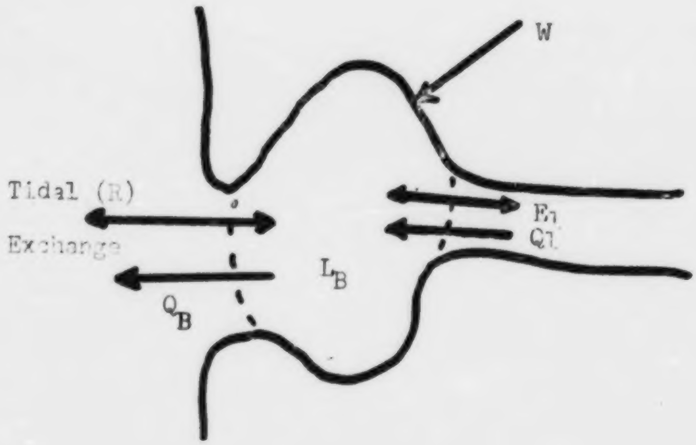
Completely Mixed Bay with Exchange

Concentration Equality:

$$L_1 = L_B$$

Mass Balance:

In this instance, the mass balance is obtained around the entire bay.



Note: VOL = Volume

Entering Bay:

$$Q_1 L_1 - E_1 A_1 \frac{dL_1}{dx} + W$$

Leaving Bay:

$$Q_B L_B + R \cdot L_B \cdot VOL + VOL \cdot L_B \cdot RK$$

Balance:

$$Q_1 L_1 - E_1 A_1 \frac{dL_1}{dx} + W = L_B Q_B + VOL (R + RK) L_B$$

Evaluating L_1 and $\frac{dL_1}{dx}$ as functions of distance and the unknown coefficients results in the required boundary equation.

An additional concept and phenomena has been introduced in the consideration of the completely mixed bay. The tidal exchange "R" provides a method of evaluating the transport of material out of the bay into the ocean by the tidal volume.

Tidal exchange "R" is defined by:

$$R = \frac{\text{tidal volume}}{\text{mean volume of the bay}} \times \frac{\text{number of tidal cycles}}{\text{day}}$$

In the simplified estuary problem used as a detailed example of boundary condition evaluation, it was possible to solve for the unknown coefficients B and C manually. As the estuary becomes more complex, it is impractical to evaluate the coefficients manually. In this latter case, it is convenient to arrange the unknown coefficients and boundary condition equations into a matrix and employ computational techniques for numerical evaluation of the unknown coefficients. The boundary condition equation (25) and (30A) from our simplified example can be arranged into matrix form:

$$\begin{bmatrix} 1 & -1 \\ EAS_1 & -EAV_2 \end{bmatrix} \begin{pmatrix} B_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 0 \\ W \end{pmatrix}$$

Matrix

Vector of
Unknown
Coefficients

Vector of
Forcing
Functions

Another way of presenting this matrix is as shown below. Both ways are used in this report.

B.C. No.	B_1	C_2	Forcing Function
3	1	-1	= 0
4	EAS_1	$-EAV_2$	= W

The basic method of establishing a matrix of unknown coefficients can be employed regardless of the complexity of the system.

There are several general considerations which are useful in evaluating the unknown coefficients and establishing matrices.

1. Evaluation of boundary conditions always results in a square matrix which means that unique solutions for the values of the unknown coefficients are available.

2. At the junction of "n" segments, there are always generated one mass balance equation and (n - 1) concentration equality equations.

3. The boundary conditions for the BOD and dissolved oxygen deficit solutions are identical with two exceptions. First, the roots of the BOD solution, S and V, are replaced by the roots of the deficit solution, SD and VD when DO deficit calculations are made. Second, in the case of boundary conditions in which flow enters or leaves a completely mixed bay, a reaction rate for the bay will appear as a part of a matrix element. This will change from the BOD removal rate (RK) when solving for BOD to the reaeration rate (AK) when solving for the deficit.

4. The right hand matrix of forcing functions are different for the BOD and dissolved oxygen deficit solution.

SAMPLE PROBLEM

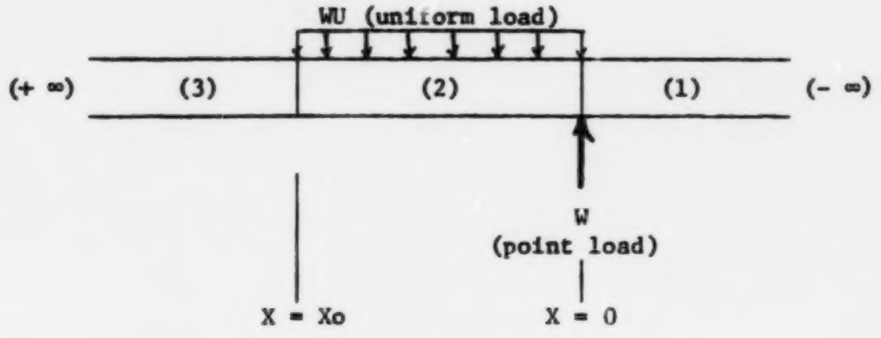


Figure 11

The purpose of this problem is to apply the theory of the previous sections to the relatively simple estuarine system shown in Figure 11.

This estuary is divided into three segments. Section 1 stretches from minus infinity to a point which will be designated as X = 0 at which a point load enters the system. Section 2 extends a distance, Xo, downstream from the zero point. A uniform waste load is input to this reach along its entire length. Finally, Section 3 extends from Xo to positive infinity.

The following equations can then be written for each section for BOD:

Section 1:

$$L_1 = B_1 e^{S_1 X} + C_1 e^{V_1 X} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (34)$$

Section 2:

$$L_2 = B_2 e^{S_2 X} + C_2 e^{V_2 X} + \frac{WU}{A_2 \cdot RK_2} \quad \ldots\ldots\ldots\ldots \quad (35)$$

Section 3:

$$L_3 = B_3 e^{S_3 X} + C_3 e^{V_3 X} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (36)$$

Six boundary conditions can then be applied:

1) $L_1 = 0$ @ $X = -\infty$

2) $L_3 = 0$ @ $X = +\infty$

Concentration equality equations:

3) $L_1 = L_2$ @ $X = 0$

4) $L_2 = L_3$ @ $X = Xo$

Mass balance equations:

5) mass balance @ $X = 0$

6) mass balance @ $X = Xo$

These six conditions may now be applied to equations 34, 35 and 36 and will provide a means of evaluating the unknown coefficients.

As shown on page 33, the infinity boundary conditions may be applied with the result that

@ $X = -\infty$ $C_1 = 0$

@ $X = +\infty$ $B_3 = 0$

Since $C_1 = 0$ the following equations result at $X = 0$

$$L_1 = B_1$$

$$L_2 = B_2 + C_2 + \frac{WU}{A_2 \cdot RK_2}$$

Boundary Condition 3 may then be applied:

$$L_1 = L_2$$

$$B_1 = B_2 + C_2 + \frac{WU}{A_2 \cdot RK_2}$$

Therefore,

$$B_1 - B_2 - C_2 = \frac{WU}{A_2 \cdot RK_2} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (37)$$

Since $B_3 = 0$ the following equations result at $X = Xo$

$$L_2 = B_2 e^{S_2 Xo} + C_2 e^{V_2 Xo} + \frac{WU}{A_2 RK_2}$$

$$L_3 = C_3 e^{V_3 Xo}$$

Applying Boundary Condition 4

$$L_2 = L_3$$

$$B_2 e^{S_2 Xo} + C_2 e^{V_2 Xo} + \frac{WU}{A_2 RK_2} = C_3 e^{V_3 Xo}$$

Therefore,

$$B_2 e^{S_2 Xo} + C_2 e^{V_2 Xo} - C_3 e^{V_3 Xo} = \frac{WU}{A_2 \cdot RK_2} \quad \ldots\ldots \quad (38)$$

As can be seen two coefficients have been eliminated (C_1 and B_3) and two equations have been generated. An additional two equations are required to obtain four equations for the four remaining unknowns (B_1, B_2, C_2, C_3). The mass balance conditions provide these equations.

Boundary condition 5 @ $X = 0$

$$Q_2 L_2 - Q_1 L_1 + E_1 A_1 \frac{dL_1}{dX} - E_2 A_2 \frac{dL_2}{dX} = W \quad \ldots\ldots \quad (39)$$

Again recall that

$$L_1 = B_1 e^{S_1 X} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (40)$$

$$L_2 = B_2 e^{S_2 X} + C_2 e^{V_2 X} \quad \ldots\ldots\ldots\ldots\ldots \quad (41)$$

Substituting 40 and 41 into 39

$$B_1(E_1A_1S_1-Q_1) + B_2(Q_2-E_2A_2S_2) + C_2(Q_2-E_2A_2V_2) = W \quad . \; . \; . \; . \; . \; . \quad (42)$$

Similarly, boundary condition number 6 can be evaluated

$$-Q_2L_2 + E_2A_2\frac{dL_2}{dx} + Q_3L_3 - E_3A_3\frac{dL_3}{dx} = 0$$

The resulting expression is

$$B_2\,[e^{S_2Xo}(E_2A_2S_2-Q_2)] + C_2\,[e^{V_2Xo}(E_2A_2V_2-Q_2)]$$

$$+ C_3\,[e^{V_3Xo}(Q_3-E_3A_3V_3)] = 0 \quad . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \quad (43)$$

Equations 37, 38, 42 and 43 now contain 4 unknowns and can be written in matrix form as in Table 3. All matrix elements can be obtained from the geometry of the system. The forcing function is known and the values of B_1, B_2, C_2 and C_3 can be obtained. By substituting them into equation 34, 35 and 36, the concentration of L can be computed for any point in the estuary.

TABLE 3

$$\begin{bmatrix} 1 & -1 & -1 & 0 \\ (E_1A_1S_1-Q_1) & (Q_2-E_2A_2S_2) & (Q_2-E_2A_2V_2) & 0 \\ 0 & e^{S_2Xo} & e^{V_2Xo} & -e^{V_3Xo} \\ 0 & e^{S_2Xo}(E_2A_2S_2-Q_2) & e^{V_2Xo}(E_2A_2V_2-Q_2) & e^{V_3Xo}(Q_3-E_3A_3V_3) \end{bmatrix} \begin{pmatrix} B_1 \\ B_2 \\ C_2 \\ C_3 \end{pmatrix} = \begin{pmatrix} \frac{WU}{A_2RK_2} \\ W \\ \frac{WU}{A_2RK_2} \\ \end{pmatrix}$$

The preceding sections have presented techniques for developing equations which define the concentration profile of BOD, dissolved oxygen deficit, and conservative substances. In addition a discussion of segmentation and methods of evaluation of boundary conditions and unknown coefficients for rather simple cases have been presented. As was illustrated in the sample problem of the previous section, the mathematical formulation of ES001 reduces to the following:

$$[A] \cdot (X) = (b) \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (44)$$

where

[A] - is the matrix of known factors

(b) - is the forcing function vector

(X) - is the vector of unknown coefficients

This relationship is formed by applying appropriate boundary conditions to the solutions to the water quality equations of the previous sections. It is solved by inverting [A] which allows the calculation of (X)

$$(X) = (b) \cdot [A]^{-1} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (45)$$

In the following sections the method of analysis to formulate equation 44 will be explained. Attention will be devoted to the actual formulation of [A] and (b) via the specific boundary conditions of ES001.

Method for Evaluation and Presentation of Boundary Conditions for Complex Estuaries:

The matrix of unknown coefficients generated for complex water quality models can become large. Presentation of the specific values of the elements of the matrix for BOD and deficit solutions would be obscure. In view of the above, and the fact that complex systems must be evaluated on a computer, the following procedure can be adopted for presentation of the water quality models.

- The BOD and dissolved oxygen deficit equations can be presented in general form.
- A table can be developed defining the specific functional forms of the general equations for the BOD and dissolved oxygen deficit solutions.
- The concentration boundary condition may be generalized in four (4) classes which cover all of the concentration equality equations encountered in the present system. The four classes of concentration equality boundary equations can be presented in a form which is amenable to matrix solution.
- The first derivative of the generalized BOD and dissolved oxygen deficit concentration profile equations can be defined in tabular form.
- Ten (10) generalized solutions to the mass balance boundary condition equations can be presented. These equations cover all of the mass balance conditions required in the present systems. The mass balance equations have been presented in a form amenable to matrix solution of the simultaneous equations developed.
- The boundary conditions may be classified and tabulated.
- The specific boundary conditions from the general classification listing can be presented in tabular form for each junction. The appropriate designation of upstream and downstream must also be indicated.
- A model can then be constructed by joining segments.

1) Presentation of equations in general form.

The formulas describing water quality in any specific segment (I) can be written in the general form:

$$L(I) = B(I)f_r(X,I) + C(I)\overline{f}_r(X,I) + YB(I) \quad . . . . (46)$$

$$D(I) = G(I)f_a(X,I) + H(I)\overline{f}_a(X,I) + YD(X,I) \quad . . . (47)$$

where:

(I) = Section number

(X,I) = Indicates that the expressions are a function of both section and location "X" in the section.

L(I) = BOD in Section I.

D(I) = Dissolved oxygen deficit in Section I.

B(I) and C(I) = Unknown coefficients for the BOD equation.

G(I) and H(I) = Unknown coefficients for the deficit equations.

$f_r(X,I)$ and $\overline{f}_r(X,I)$ = Functional forms for the BOD equation. The specific forms are defined in Table 4.

YB(I) = Particular integral for the BOD solution as defined in Table 4.

YD(X,I) = Particular integral for the deficit solution as defined in Table 4.

$f_r(X,I)$, $\overline{f}_r(X,I)$, $f_a(X,I)$, and $\overline{f}_a(X,I)$ are functions of distances X and segment number "I". YB(I) is a function of the segment number "I" alone, and YD(X,I) is a function of segment and location.

There are three (3) possible special cases which do not have the form indicated above. They are:

1. Negative infinity boundary conditions where the only unknown coefficient is the B or G coefficient.

$$L(I) = B(I)f_r(X,I) + YB(I) \quad \ldots\ldots\ldots\ldots \quad (48)$$

$$D(I) = G(I)f_a(X,I) + YD(I) \quad \ldots\ldots\ldots\ldots \quad (48a)$$

2. Positive infinity boundary conditions where the only unknown coefficient is the C or H coefficient.

$$L(I) = C(I)\bar{f}_r(X,I) + YB(I) \quad \ldots\ldots\ldots\ldots \quad (49)$$

$$D(I) = H(I)\bar{f}_a(X,I) + YD(I) \quad \ldots\ldots\ldots\ldots \quad (49a)$$

3. Completely mixed volumes.

$$L(I) = B(I) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (50)$$

$$D(I) = G(I) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (50a)$$

For completely mixed volumes the BOD and deficit does not vary with distance and can therefore be defined by constants B(I) and G(I) respectively.

2) Specific functional forms of the general equations for the BOD and D.O. deficit solution.

Function	Value
$f_r(X,I)$	$e^{S(I)X}$
$\overline{f}_r(X,I)$	$e^{V(I)X}$
$f_a(X,I)$	$e^{SD(I)X}$
$\overline{f}_a(X,I)$	$e^{VD(I)X}$
$YB(I)$	$\frac{WU(I)}{A(I)\cdot RK(I)}$
$YD(X,I)$	$\frac{BD(I)}{HT(I)\cdot AK(I)} + \frac{RA(I)}{AK(I)}$ $+ \frac{DK(I)\cdot WU(I)\cdot F(I)}{AK(I)\cdot RK(I)\cdot A(I)}$ $+ \frac{DK(I)\cdot F(I)}{AK(I)-RK(I)} \; [B(I)f_r(X,I)$ $+ C(I)\overline{f}_r(X,I)]$

TABLE 4

3) Concentration Equality Equations

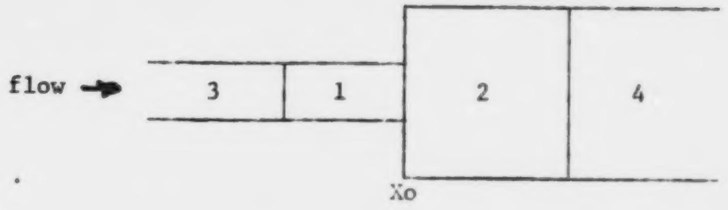


Figure 12: One Dimensional Estuary

To illustrate the formulation of the concentration boundary conditions in computational form the following example has been included.

Figure 12 is a one dimensional estuary with a change in cross-sectional area at Xo. According to the concentration equality boundary condition for two double sections

$$L_1 = L_2 \quad @ \ X = X_o$$

$$B(1)\ f_r(X_o,1) + C(1)\ \overline{f}_r(X_o,1) + YB(1) =$$

$$B(2)\ f_r(X_o,2) + C(2)\ \overline{f}_r(X_o,2) + YB(2)$$

Group terms:

$$B(1)\ f_r(X,1) + C(1)\ \overline{f}_r(X,1) - B(2)\ f_r(X,2)$$

$$- C(2)\ \overline{f}_r(X,2) = YB(2) - YB(1)$$

Consulting Table 4 we can then formulate the following matrix denoting it as Boundary Condition "J"

BC#	B(1)	C(1)	B(2)	C(2)
J	$e^{S(1)Xo}$	$e^{V(1)Xo}$	$-e^{S(2)Xo}$	$-e^{V(2)Xo}$

with the forcing function for BOD

$$BFORC(J) = YB(2) - YB(1)$$

for DO deficit the matrix would be the same but with f_a and $\overline{f}_a$ replacing f_r and $\overline{f}_r$ and

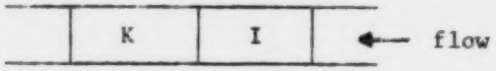
$$DFORC(J) = YD(X,2) - YD(X,1)$$

Boundary condition "J" can also be expressed in the following way substituting the functional values shown below

B(1)	$f_r(X,1)$
C(1)	$\overline{f}_r(X,1)$
B(2)	$f_r(X,2)$
C(2)	$\overline{f}_r(X,2)$

Now, using the above notation, the boundary conditions of ES001 can be generalized. They will be set up for BOD with additional notes to illustrate the procedure for DO deficit.

CLASS 1 - Boundary Condition - (Concentration Equality Between Two Double Sections*);



Elements of matrix for Sections I and K and Boundary Condition J

B(I)	$f_r(X,I)$
C(I)	$\overline{f}_r(X,I)$
B(K)	$-f_r(X,K)$
C(K)	$-\overline{f}_r(X,K)$

---

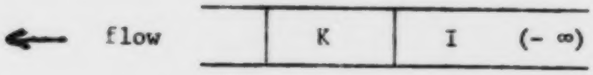
* Note - a double section is defined as a section which is bounded on both ends by another section.

BFORC(J) = YB(K) - YB(I)

DFORC(J) = YD(X,K) - YD(X,I)

For deficit Matrix, replace f_r and $\overline{f}_r$ by f_a and $\overline{f}_a$, respectively.

CLASS 2 - Boundary Condition (Concentration Equality, Negative Infinity Boundary for Section I, with C(I) = 0)

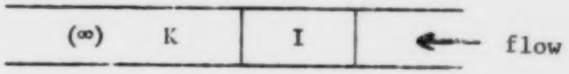


Same as Class 1 Boundary Condition, but with

C(I) | 0

Forcing functions are the same as for Class 1 Boundary Condition.

CLASS 3 - Boundary Condition (Concentration Equality, Positive Infinity Boundary for Section K, with B(K) = 0)

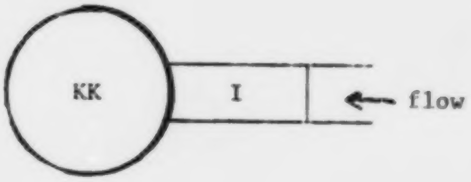


Same as Class 1 Boundary Condition, with

B(K) | 0

Forcing functions are the same as for Class 1 Boundary Condition.

CLASS 4 - Boundary Condition (Concentration Equality, between a double section and a completely mixed volume)



$$\left.\begin{array}{l}B(I)\\ C(I)\\ B(KK)\end{array}\right| \begin{array}{l}= f_r(X,I)\\ = \overline{f}_r(X,I)\\ = -1\end{array}$$

$$BFORC(J) = YB(I)$$

$$DFORC(J) = YD(X,I)$$

For deficit matrix replace, f_r and $\overline{f}_r$ with f_a and $\overline{f}_a$.

4) Specific functional forms of the derivatives according to x of the general equation of the BOD and DO deficit solution.

Function	Value
$f'_r(X,I)$	$S(I)e^{S(I)X}$
$\overline{f}'_r(X,I)$	$V(I)e^{V(I)X}$
$f'_a(X,I)$	$SD(I)e^{SD(I)X}$
$\overline{f}'_a(X,I)$	$VD(I)e^{VD(I)X}$
$YB'(I)$	Zero
$YD'(X,I)$	$\frac{DK(I) \cdot F(I)}{AK(I)-RK(I)}\,[B(I)f'_r(X,I) + C(I)\overline{f}'_r(X,I)]$

TABLE 5

5) Mass Balance Equation

Additional classes of boundary conditions are required for the mass balances. Before evaluation of mass balance boundary conditions, it is necessary to define the first derivatives of the concentration profile equations.

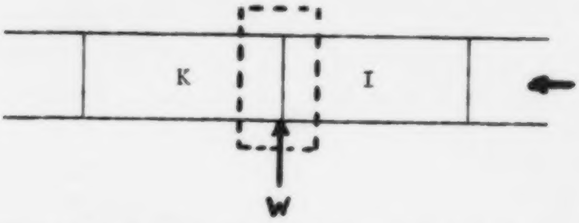
$$\frac{dL}{dx}(I) = B(I)f_r'(X,I) + C(I)\overline{f}_r'(X,I) + YB'(I)$$

$$\frac{dD}{dx}(I) = G(I)f_a'(X,I) + H(I)\overline{f}_a'(X,I) + YD'(X,I)$$

Table 5 presents the definitions of the functions f_r', $\overline{f}_r'$, f_a', $\overline{f}_a'$, YD', and YD' used in the present models.

Generalized solutions are available for the mass balances when sections I, K, and KK are considered and the boundary equation number is defined by J. It must be noted that Section I is defined as upstream, or more negative than Section K. (That is, flow from Section I enters Section K.) The direction of Section KK must be indicated.

CLASS 5 - Boundary Condition (Mass Balance,
Two Adjacent Sections)



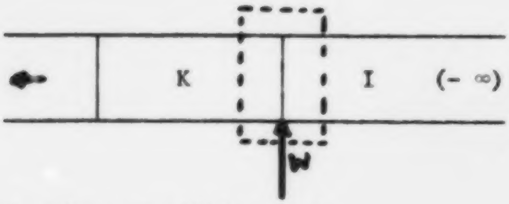
NOTE: The portion enclosed by the dotted lines in the figure denote the element about which the mass balance is obtained as on page 35.

B(I)	$E(I)\cdot A(I)\cdot f_r'(X,I) \;-\; Q(I)f_r(X,I)$
C(I)	$E(I)\cdot A(I)\cdot \overline{f}_r'(X,I) \;-\; Q(I)\overline{f}_r(X,I)$
B(K)	$Q(K)f_r(X,K) \;-\; E(K)\cdot A(K)\cdot f_r'(X,K)$
C(K)	$Q(K)\overline{f}_r(X,K) \;-\; E(K)\cdot A(K)\cdot \overline{f}_r'(X,K)$

BFORC(J) = W(n) (load entering at junction) + Q(I)YB(I) - Q(K)YB(K)

$$\mathrm{DFORC}(J) = Q(I)YD(X,I) - E(I)\cdot A(I)\cdot YD'(X,I) + E(K)\cdot A(K)\cdot YD'(X,K) - Q(K)YD(X,K)$$

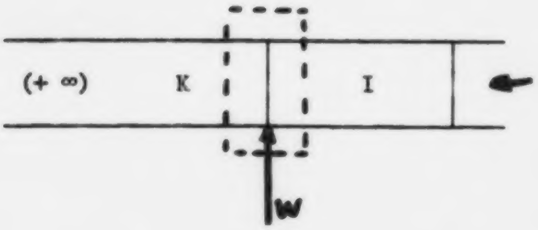
CLASS 6 - Boundary Condition (Mass Balance with Negative Infinity Boundary for Section I makes C(I) = 0)



Same as for Class 5 Boundary Condition with

C(I)	0

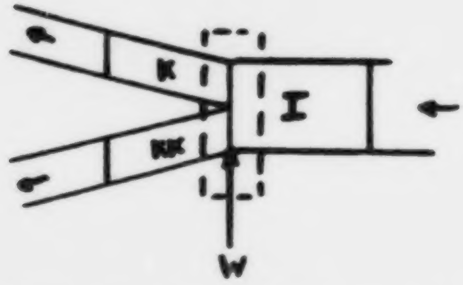
CLASS 7 - Boundary Condition (Mass Balance with Positive Infinity Boundary for Section K makes B(K) = 0)



Same as for Class 5 Boundary Condition with

B(K)	0

CLASS 8 - Boundary Condition (Mass Balance with Flow Entering Sections K and KK from Section I)



C(I), B(I), C(K), and B(K) are same as for Class 5 Boundary Condition.

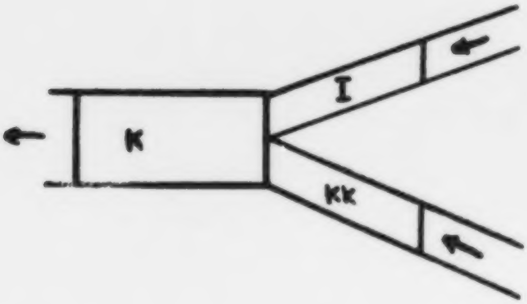
B(KK)	$Q(KK)f_r(X,KK) - E(KK)\cdot A(KK)\cdot f'_r(X,KK)$
C(KK)	$Q(KK)\overline{f}_r(X,KK) - E(KK)\cdot A(KK)\cdot \overline{f}'_r(X,KK)$

$BFORC(J) = W(N) + Q(I)YB(I) - Q(K)YB(K) - Q(KK)YB(KK)$

DFORC(J) = DFORC(J) in Class 5 Boundary, plus:

$E(KK)\cdot A(KK)\cdot YD'(X,KK) - Q(KK)YD(X,KK)$

CLASS 9 - Boundary Condition (Mass Balance with Flow Entering Section K from Sections I and KK)



B(I), C(I), B(K) and C(K) are the same as for Boundary Condition 5.

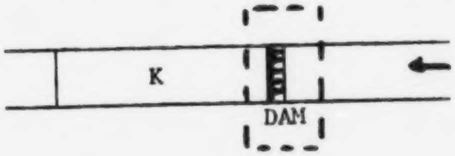
$$\left.\begin{array}{l} \mathrm{B(KK)} \\[1ex] \mathrm{C(KK)} \end{array}\right| \begin{array}{l} E(KK)\cdot A(KK)\cdot f_r'(X,KK) - Q(KK)\cdot f_r(X,KK) \\[1ex] E(KK)\cdot A(KK)\cdot \bar{f}_r'(X,KK) - Q(KK)\cdot \bar{f}_r(X,KK) \end{array}$$

$$\mathrm{BFORC}(J) = W(N) + Q(I)YB(I) + Q(KK)YB(KK) - Q(K)YB(K)$$

DFORC(J) = DFORC(J) as in Class 5 boundary condition, plus:

$$Q(KK)\cdot YD(X,KK) - E(KK)\cdot A(KK)\cdot YD'(X,KK)$$

CLASS 10 - Boundary Condition (Mass Balance at Section with Advective Flux, only) - DAM



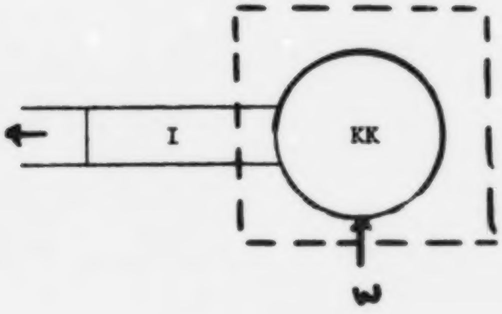
$$\left.\begin{array}{l} \mathrm{B(K)} \\[1ex] \mathrm{C(K)} \end{array}\right| \begin{array}{l} Q(K)\cdot f_r(X,K) - E(K)\cdot A(K)\cdot f_r'(X,K) \\[1ex] Q(K)\cdot \bar{f}_r(X,K) - E(K)\cdot A(K)\cdot \bar{f}_r'(X,K) \end{array}$$

$$\mathrm{BFORC}(J) = Q(DAM)\cdot L(DAM) - Q(K)YB(K)$$

$$\mathrm{DFORC}(J) = E(K)\cdot A(K)\cdot YD'(X,K) - Q(K)\cdot YD(X,K) + DEF(DAM)\cdot Q(DAM)$$

L(DAM) and DEF(DAM) equal the concentration of BOD and dissolved oxygen deficit in the flow over a dam while Q(DAM) is the flow over the dam.

CLASS 11 - Boundary Condition (mass Balance with Section I Leaving Completely Mixed Volume, KK)



$$B(I) = Q(I)\cdot f_r(X,I) - E(I)\cdot A(I)\cdot f_r'(X,I)$$

$$C(I) = Q(I)\cdot \overline{f}_r(X,I) - E(I)\cdot A(I)\cdot \overline{f}_r'(X,I)$$

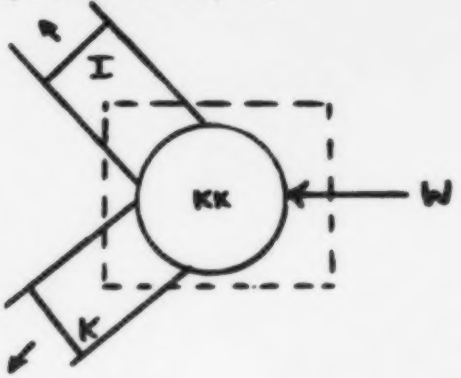
$$B(KK) = VOL(KK)\cdot RK(KK)$$

$$BFORC(J) = W \text{ (bay load)} - Q(I)YB(I)$$

$$DFORC(J) = E(I)\cdot A(I)\cdot YD'(X,I) - Q(I)\cdot YD(X,I) + F(KK)\cdot DK(KK)B(KK)\cdot VOL(KK)$$

For DO deficit change RK(KK) of B(KK) to AK(KK). Note that a load should not be thought of as entering at the junction in this case. This is so for all the bays since the balance is taken about the entire bay and all loads are assumed to flow directly into the bay.

CLASS 12 - Boundary Condition (Mass Balance with Sections I and K Leaving Completely Mixed Volume, KK)



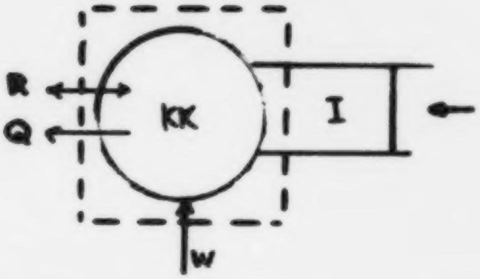
For two sections leaving a completely mixed volume, add B(K) and C(K) (which are identical in form to B(I) and C(I) of the Class 11 Boundary Condition)

To the forcing functions, add:

$BFORC(J)$ plus $-\ Q(K)\cdot YB(K)$

$DFORC(J)$ plus $\quad E(K)\cdot A(K)\cdot YD'(X,K) \ - \ Q(K)YD(X,K)$

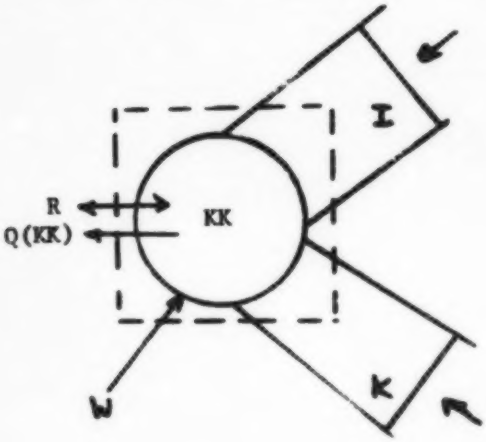
CLASS 13 - Boundary Condition (Mass Balance, Section I Entering a Completely Mixed Volume, KK, with a Tidal Exchange and Flow from the Volume to an Infinite Medium)



B(I)	$E(I)\cdot A(I)f_r'(X,I) \ - \ Q(I)f_r(X,I)$
C(I)	$E(I)\cdot A(I)\bar{f}_r'(X,I) \ - \ Q(I)\bar{f}_r(X,I)$
B(KK)	$(Q(KK) + R(KK)VOL(KK) + VOL(KK)\cdot RK(KK))$
BFORC(J)	$W(KK) + Q(I)YB(I)$
DFORC(J)	$Q(I)YD(X,I) - E(I)\cdot A(I)\cdot YD'(X,I) + F(KK)\cdot DK(KK)B(KK)\cdot VOL(KK)$

For DO deficit change R(KK) of AM(KK,J) to AK(KK).

CLASS 14 - Boundary Condition (Mass Balance, Sections I and K Entering a Completely Mixed Volume, KK, with a Tidal Exchange, R, and Flow from the Volume to the Infinite Medium)



For two sections I and K, entering a completely mixed volume, add B(K) and C(K), which are identical in form to B(I) and C(I) to the Class 13 Boundary Condition.

To the Forcing Functions, add:

BFORC(J) plus + Q(K)YB(K)

DFORC(J) plus Q(K)YD(X,K) - E(K)A(K)YD(X,K)

6) Tabulation of Boundary Condition

The boundary condition classifications applicable to each combination of sections are described in Table 6.

Using Table 6 the specific boundary condition can be tabulated for each junction in the system. A model can then be constructed by joining segments.

TABLE 6

BOUNDARY CLASSIFICATIONS

Boundary Classification	Types of Applications
1	Concentration equality between two double sections (when both sections have two (2) unknown coefficients and are bounded at both ends.)
2	Concentration equality between a double and a negative infinity section (when one section has two (2) unknown coefficients and the second section has only B or G coefficient.)
3	Concentration equality between a double and a positive infinity section (when one section has two (2) unknown coefficients and the second section has only C or H coefficient.)
4	Concentration equality between a double section and a completely mixed volume.
5	Mass balance between two double sections, both sections having two (2) unknown coefficients.
6	Mass balance between a double section and a negative infinity section when one section has two (2) unknown coefficients and the second section has only B or G coefficients.)
7	Mass balance between a double section and a positive infinity section (when one section has two (unknown) coefficients and the second section has only a C or H coefficient.
8	Mass balance for three (3) double sections with flow from one section dividing into the two sections.
9	Mass balance for three (3) double sections with flow from two sections combining into the third section.

TABLE 6 (Continued)

BOUNDARY CLASSIFICATIONS

Boundary Classification	Types of Applications
10	Mass balance at a section with only advective flux entering (dam).
11	Mass balance between a double section and a completely mixed volume. Flow leaves the bay through the section.
12	Mass balance between two (2) double sections and a completely mixed volume. Flow leaves the bay through the sections.
13	Mass balance between a completely mixed volume and a double section when flow enters bay through the section.
14	Mass balance between a completely mixed volume and two (2) double sections when flow enters bay through two sections.

NOMENCLATURE

Symbol	Definition	Units
A	Cross-sectional area	$[L^2]$
AK	Reaeration coefficient	$[1/T]$
B	BOD constant (B associated with S)	$[M/L^3]$
BD	Bottom oxygen demand	$[M/L^2/T]$
BOD	Biochemical Oxygen Demand	$[M/L^3]$
c	Concentration of some mass	$[M/L^3]$
C	BOD constant (C associated with V)	$[M/L^3]$
CBOD	Carbonaceous Component of BOD	$[M/L^3]$
C_s	Saturation concentration of dissolved oxygen in water at a particular temperature and solids concentration	$[M/L^3]$
D	Concentration of DO deficit	$[M/L^3]$
D_g	General solution to the DO deficit concentration equation	$[M/L^3]$
DK	Deoxygenation coefficient	$[1/T]$
DO	Dissolved Oxygen	$[M/L^3]$
E	Dispersion coefficient	$[L^2/T]$
F	Ultimate to 5-day BOD ratio	-
G	DO deficit constant (G associated with S)	$[M/L^3]$
H	DO deficit constant (H associated with V)	$[M/L^3]$
HT	Water depth	$[L]$
J	Conservative substance constant	$[M/L^3]$
L	Concentration of BOD	$[M/L^3]$
L_b	Concentration of BOD in bay	$[M/L^3]$
L_d	Concentration of BOD in flow over a dam	$[M/L^3]$
L_g	General solution to the BOD concentration differential equation	$[M/L^3]$

Symbol	Definition	Units
L_p	Particular solution to the BOD concentration differential equation (same as YB)	$[M/L^3]$
L_x	Ultimate BOD of a water	$[M/L^3]$
NBOD	Nitrogenous component of BOD	$[M/L^3]$
Q	Net advective flow of an estuary	$[L^3/T]$
Q_b	Net advective flow out of a bay	$[L^3/T]$
Q_d	Flow over a dam	$[L^3/T]$
P	Algal photosynthesis effect	$[M/L^3/T]$
R	Tidal exchange coefficient	[1/T]
RA	Net algal photosynthesis and respiration effect	$[M/L^3/T]$
RK	Removal coefficient of BOD	[1/T]
RS	Algal respiration effect	$[M/L^3/T]$
$\pm\Sigma S$	Sum of sources and sinks	$[M/L^3/T]$
S	Argument of exponential in BOD solution associated with B $\frac{U}{2E}\left[1 + \left[1 + \frac{4RK \cdot E}{U^2}\right]^{1/2}\right]$	[1/L]
SD	Argument of exponential in DO deficit solution associated with G $\frac{U}{2E}\left[1 + \left[1 + \frac{4AK \cdot E}{U^2}\right]^{1/2}\right]$	[1/L]
t	Time	[T]
U	Net advective velocity	[L/T]
V	Argument of exponential in BOD solution associated with C $\frac{U}{2E}\left[1 - \left[1 + \frac{4RK \cdot E}{U^2}\right]^{1/2}\right]$	[1/L]

Symbol	Definition	Units
VD	Argument of exponential in DO deficit solution associated with H $\frac{U}{2E} \left[1 - \left[1 + \frac{4AK \cdot E}{U^2}\right]^{1/2}\right]$	[1/L]
VOL	Volume	$[L^3]$
W	Point waste load	[M/T]
WI	Width	[L]
WU	Uniform waste load	[M/L/T]
x	Distance along an estuary in direction of flow	[L]
Y_b	Sources and sinks of BOD	$[M/L^3/T]$
YB	Particular integral of BOD concentration differential equation	$[M/L^3]$
Y_c	Sources and sinks of conservative substances	$[M/L^3/T]$
YC	Particular integral of conservative concentration differential equation	$[M/L^3]$
Y_d	Sources and sinks of DO deficit	$[M/L^3/T]$
YD	Particular integral of DO deficit differential equation	$[M/L^3]$
Z	Conservative substance constant	$[M/L^3]$
ζ	Concentration of conservative substance	$[M/L^3]$

SENSITIVITY

System sensitivity may be defined as the relative change in water quality associated with changes in the parameters or the geometry of the system. There are three (3) significant areas where system sensitivity has a bearing on the use and application of mathematical models for water quality.

In the first instance, the relative sensitivity of the system to changes in the value of a parameter can be employed to determine the accuracy with which this parameter should be measured. The accuracy with which a parameter should be measured will in turn dictate the amount of money and effort to be spent in conducting field tests. As an example, a system may be relatively insensitive to fresh water flow and dispersion. Efforts to develop refined estimates of these parameters can be minimized. If, on the other hand, the system is sensitive to the nature and magnitude of waste inputs and to the value of the reaeration coefficient, expenditures to develop realistic and well-defined estimates of these parameters and inputs are justified. In summary, information on the value of all system parameters, inputs and geometry are required. Expenditures for developing highly refined estimates of the various items should be viewed in terms of their influence on the water quality profiles; that is, the systems' sensitivity.

A second area where knowledge of the system sensitivity is important is in developing a realistic set of consistent parameter values. It is necessary to obtain calculated water quality profiles which are in agreement with observed data in order to validate the model. Knowledge of the influence of changes in parameters on calculated profiles will assist in establishing the direction and magnitude of changes in the assigned value of parameters and inputs. This will speed up the process of determining a consistent set of parameters which will provide agreement between calculated and observed profiles at various temperature and flow conditions.

The third area where knowledge of system sensitivity is significant is in selection and evaluation of the methods for improving water quality. As an example, if water quality is insensitive to changes in fresh water flow, this would indicate that low flow augmentation may not be a realistic method of improving water quality. On the other hand, the system may be sensitive to tidal exchange and changes in system volume. This indicates that consideration should be given to methods of increasing tidal exchange and increasing the system volume as a means of improving water quality.

It should be noted that the sensitivity of the system to a change in the value of a single parameter will vary depending on the specific value of all parameters; several general indications of system sensitivity can be defined. These are:

1. The linear nature of the mathematical models results in changes in the calculated BOD and dissolved oxygen deficit profiles in direct proportion to changes in waste inputs. That is, a 10 percent reduction in a waste discharge load will result in a 10 percent reduction in the calculated BOD and dissolved oxygen deficit profiles associated with the load.

2. Steep concentration gradients will tend to be destroyed by relatively small increases in the value of the dispersion coefficient. It should be noted that the influence of changes in the dispersion coefficient "E" are reduced when relatively flat profiles (low concentration gradients) are present.

3. The net influence on water quality resulting from changes in the value of specific parameters will vary depending upon the location of the point in the system being considered.

To illustrate this point, Figures 13 to 17 are presented. The Figures contain plots of a plane formed by the variable fresh water velocity "U", the dispersion coefficient "E", and the calculated dissolved oxygen deficit, "D". The values of RK and AK are constant and each Figure illustrates the variation of the calculated dissolved oxygen deficit at a specific location for various combinations of values of "U" and "E". In Figures 13 to 15, for locations upstream of the point of the waste discharge, one notes that the maximum deficit always occurs at low values of "U" and "E". The shape and magnitude of the changes in curvature of the planes describe the influence of changing the parameters "U" and "E". Figures 16 to 17 are the comparable planes for locations downstream from the point of waste introduction. It will be noted from these Figures that for a constant value of the dispersion coefficient, "E", the fresh water velocity "U" at which the maximum deficit occurs increases as the location studied becomes more remote from the waste input location. That is, at a station located 10 miles from the point of waste introduction, the dissolved oxygen deficit increases as the fresh water velocity increases, with the maximum deficit occurring at the station when the flow velocity is in the order of two (2) miles per day.

The specific planes presented in this report were developed from consideration of an infinite estuary. The geometry of the estuary may have significant effects on the shape of the planes and in general, on the sensitivity and behavior of the system.

An example of the influence of geometry on system sensitivity and behavior is illustrated in Figures 18 and 19. In these Figures, a four (4) section system is considered with no fresh water flow. In Figure 18, the point of waste introduction is located between the two middle sections at X = 0. Case #1 on this Figure illustrated the calculated dissolved oxygen deficit curve when each of the four sections has the same cross-sectional area. In Case #2 of this Figure, the cross-sectional area of the extreme section has been increased by a factor of four (4). It can be observed that for Case #2, the entire deficit profile is significantly reduced in magnitude and the maximum deficit occurs behind the point of waste introduction, and away from the section with the large cross-sectional area. Figure 19 presents profiles for comparable system geometries. In this instance, the waste input location is bounded by an infinity section. It should be noted that for Case #1, in both Figures 18 and 19 when the estuary geometry is comparable in all sections, the location of the waste input does not influence the magnitude of the maximum deficit, which is 5 ppm and occurs at the point of waste input. In Case #2 for both Figures 18 and 19, the maximum deficit occurs behind the waste input, away from the section with the large cross-sectional area. Note that the magnitude of the maximum deficit is lower in Case #2, as the waste load location approaches the section with the large cross-sectional area.

Several points of information on sensitivity can be made from an analysis of the information presented in Figures 18 and 19

1. The maximum dissolved oxygen deficit occurs at the point of waste discharge for symmetrical systems with zero fresh water flow.

2. The maximum deficit need not occur at the point of waste discharge for unsymmetrical systems with zero flow.

3. The calculated dissolved oxygen deficit from a constant waste input decreases as the liquid volume of the system increases.

Figure 20 presents a plot of the plane relating the deficit to the deoxygenation "DK" and reaeration "AK" coefficient. Note the marked reduction in deficit associated with increasing reaeration coefficient. L'Hospital's rule is applicable to the range where AK = RK as the particular integral in the deficit solution contains the difference in the two coefficients (DK/AK-RK). The increase in deficit due to an increase in the deoxygenation coefficient is also indicated on the Figure. The range of the deoxygenation coefficient DK on Figure 20 is from 0.05 to 0.5 per day.

The previous discussions have indicated in a limited manner, the response of estuary systems to changes in parameters, inputs and geometry. The area of system sensitivity is relatively complex and requires additional effort before adequate definitions are available to define the anticipated behavior of the waterways which are being studied in the present program.

It is suggested that as part of the analysis and use of the mathematical models for water quality a systemic effort be made to develop an understanding of the behavior and sensitivity of the system. Specifically, it is suggested that for each of the models the following program be developed:

1. Obtain first, best estimates of all system parameters, inputs and geometry.

2. Using a unit load in all sections, develop profiles employing the best estimates obtained in No. 1 above.

3. Repeat the procedure in item No. 2 for ranges of all the system parameters. The best estimates may initially be halved and doubled. (As an example, E = 10 best estimate, also evaluate E = 20 and E = 5.)

A major challenge remains in developing a convenient graphical method of presenting and displaying this information on system behavior and sensitivity. This report has employed individual profiles and three-dimensional graphs. In all cases, room for improvement in techniques for displaying system sensitivity remain. The program for evaluating system sensitivity should include provisions to provide information in all three of the significant areas of application. Namely:

1. Determination of the variables and parameters which require refined field measurements because the system is sensitive to these items. This will provide information on what must be measured, and the required accuracy.

2. Determination of changes in calculated profiles to assist in developing a consistent set of parameters defining all of the phenomena having a major influence on water quality.

3. Determination of the feasibility of alternate methods of improving water quality based upon the systems' response to changes in parameters, inputs, and geometry.

SENSITIVITY PLANES

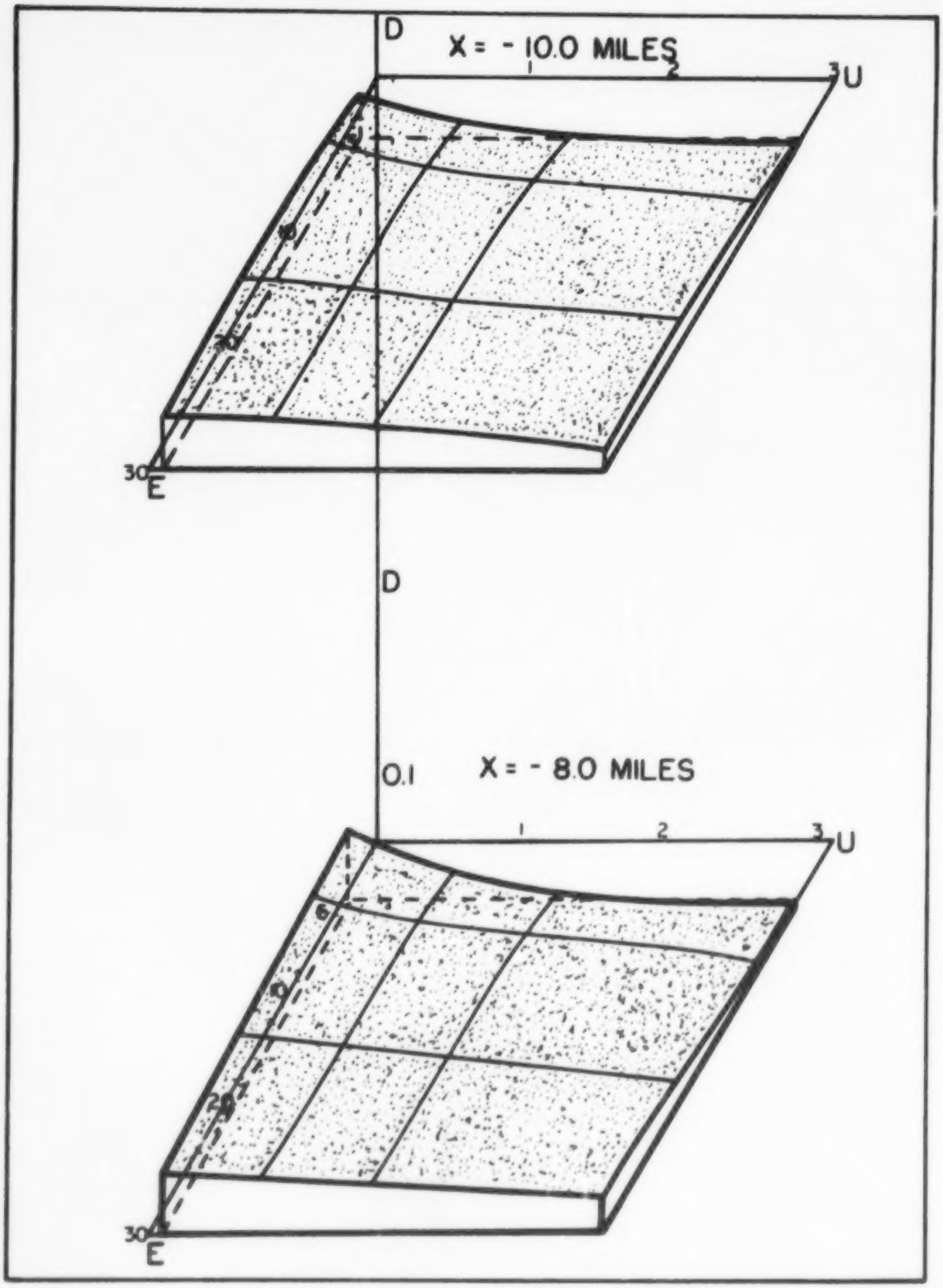


FIGURE 13

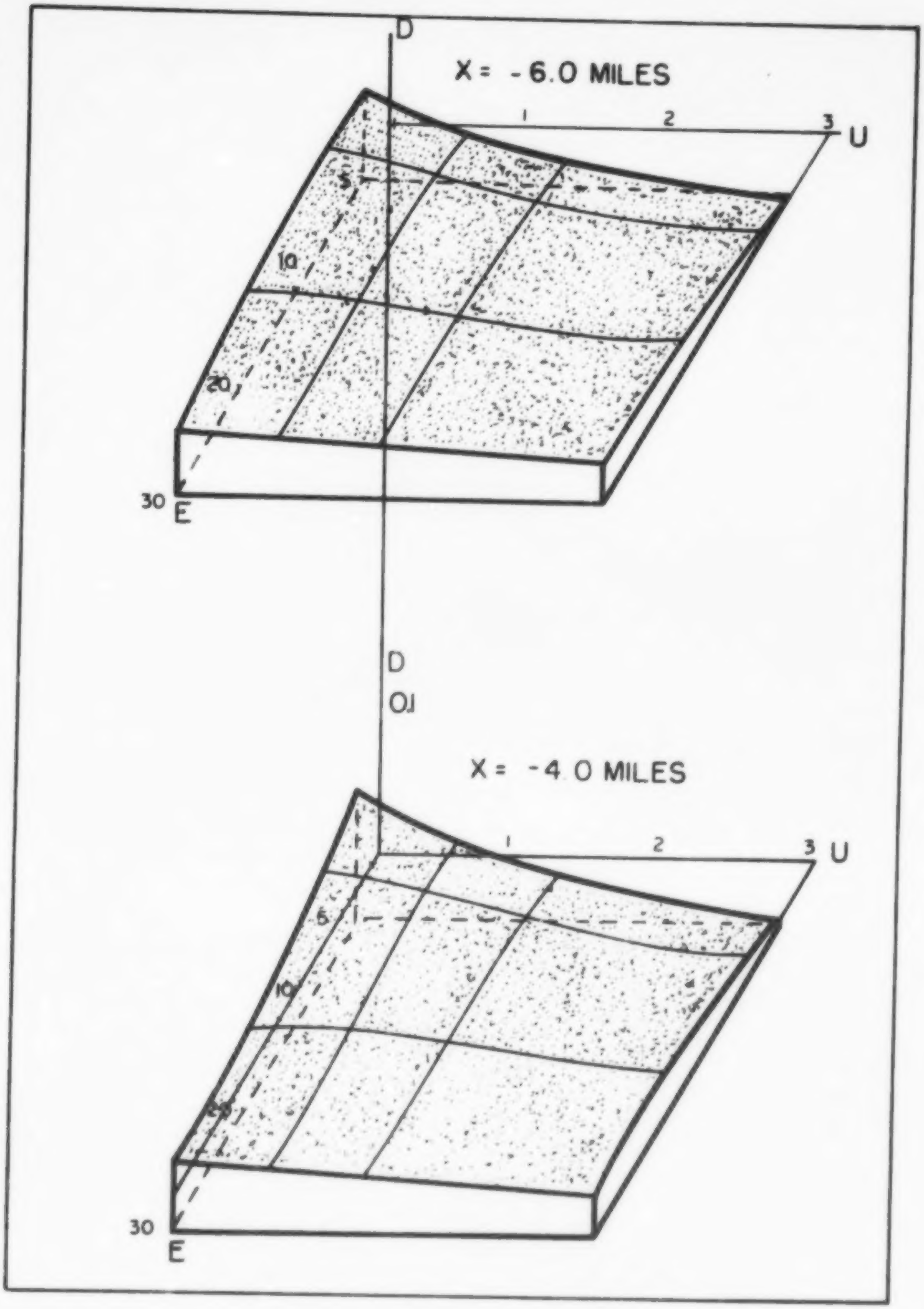


FIGURE 14

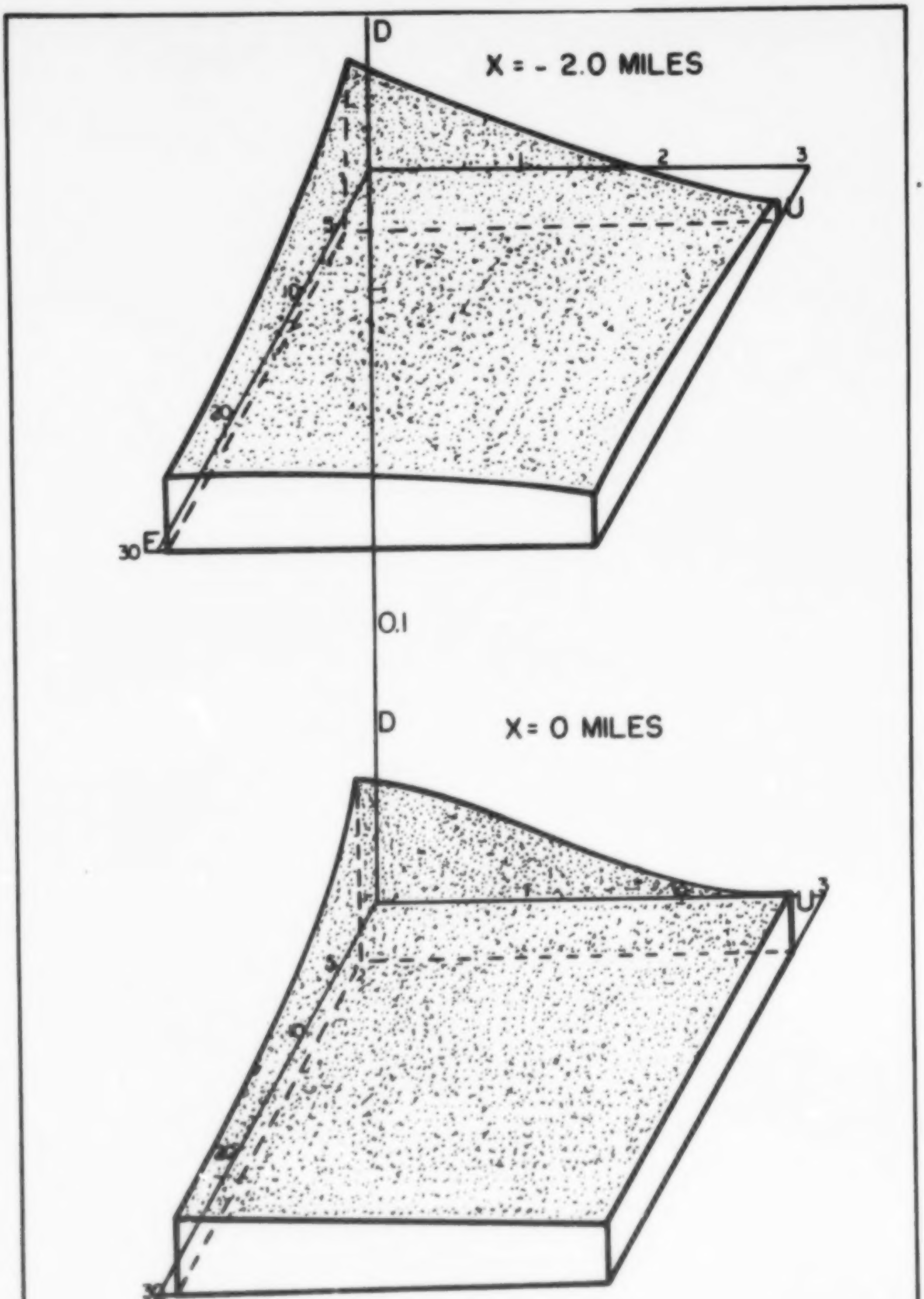


FIGURE 15

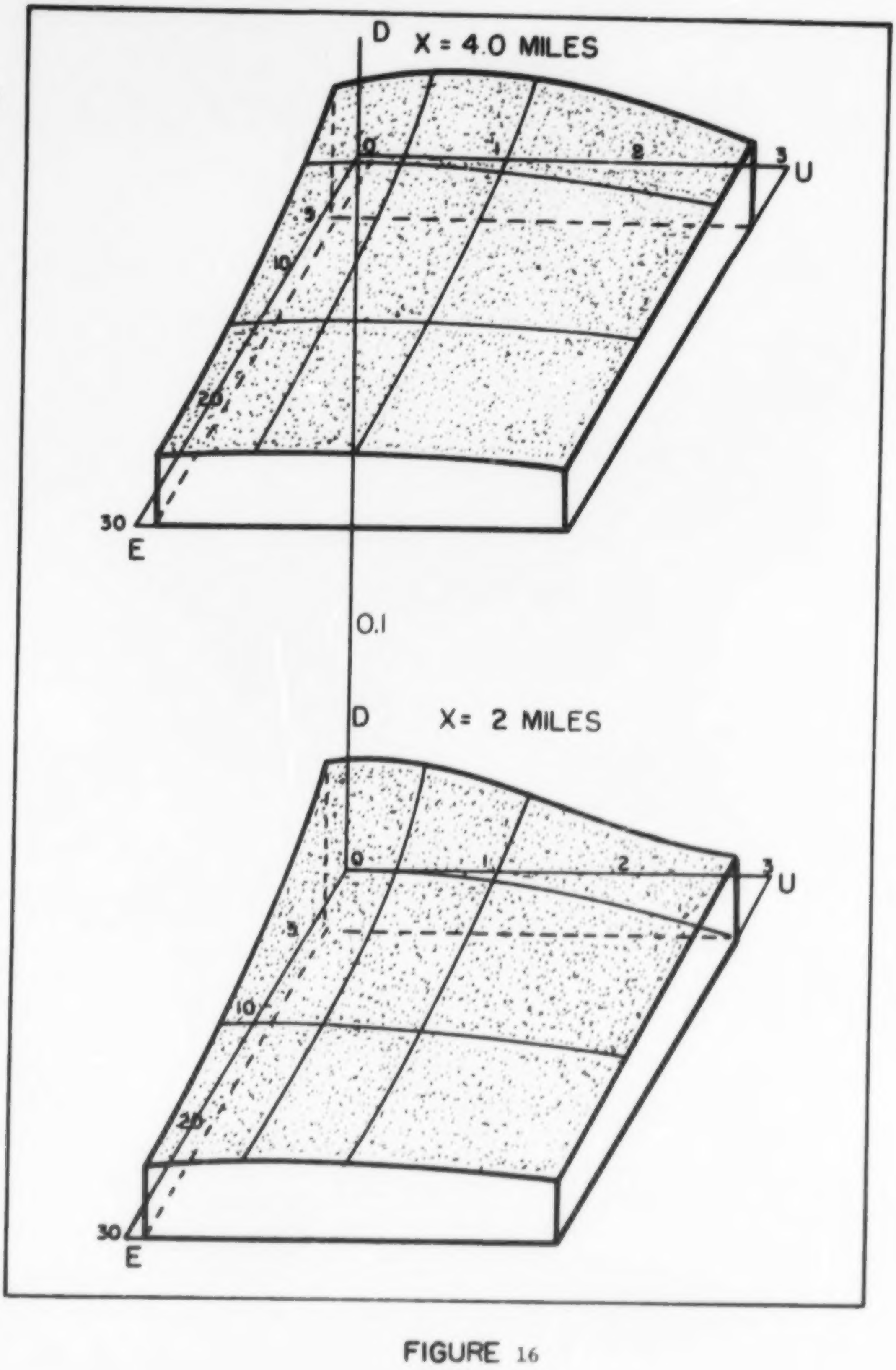


FIGURE 16

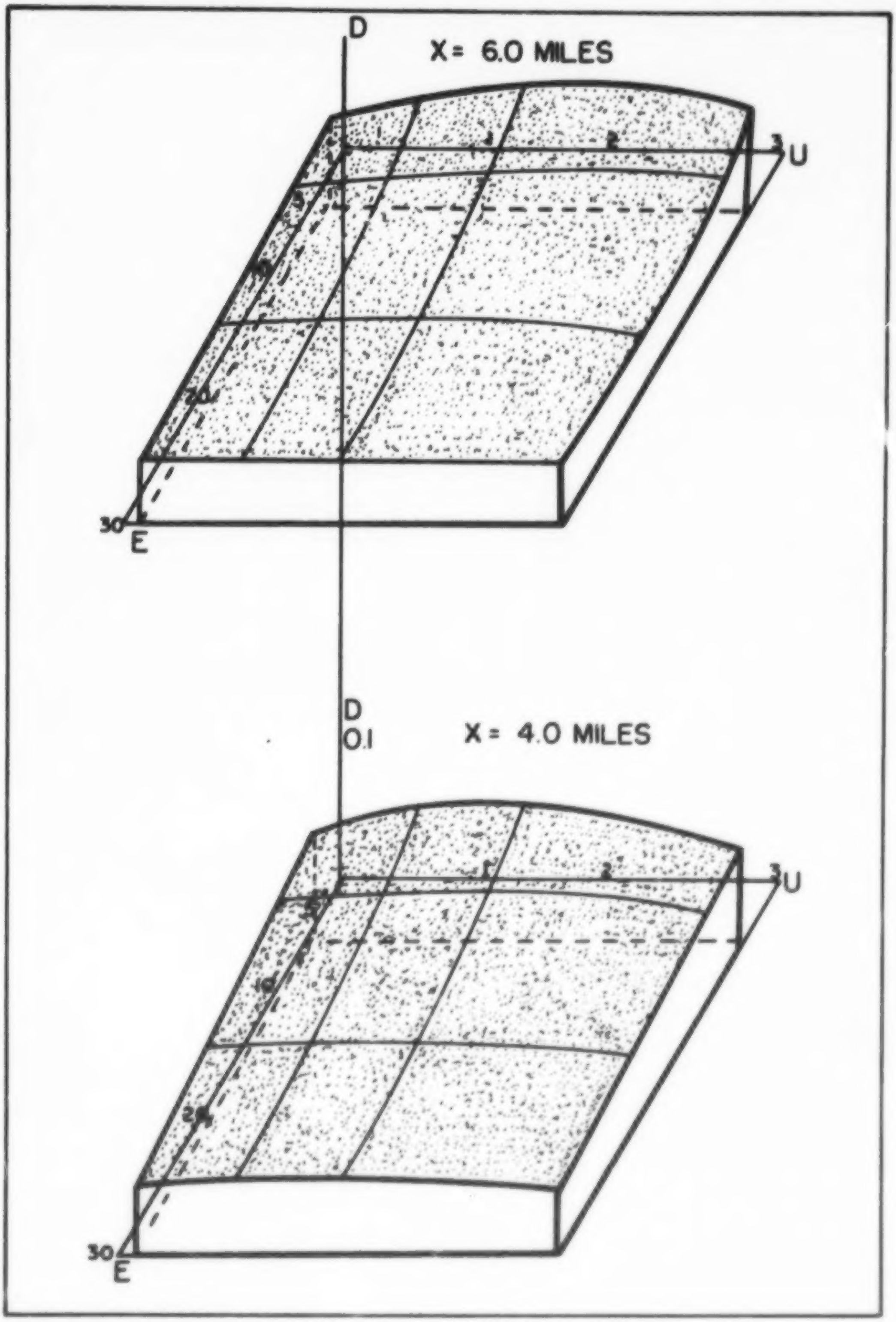


FIGURE 17

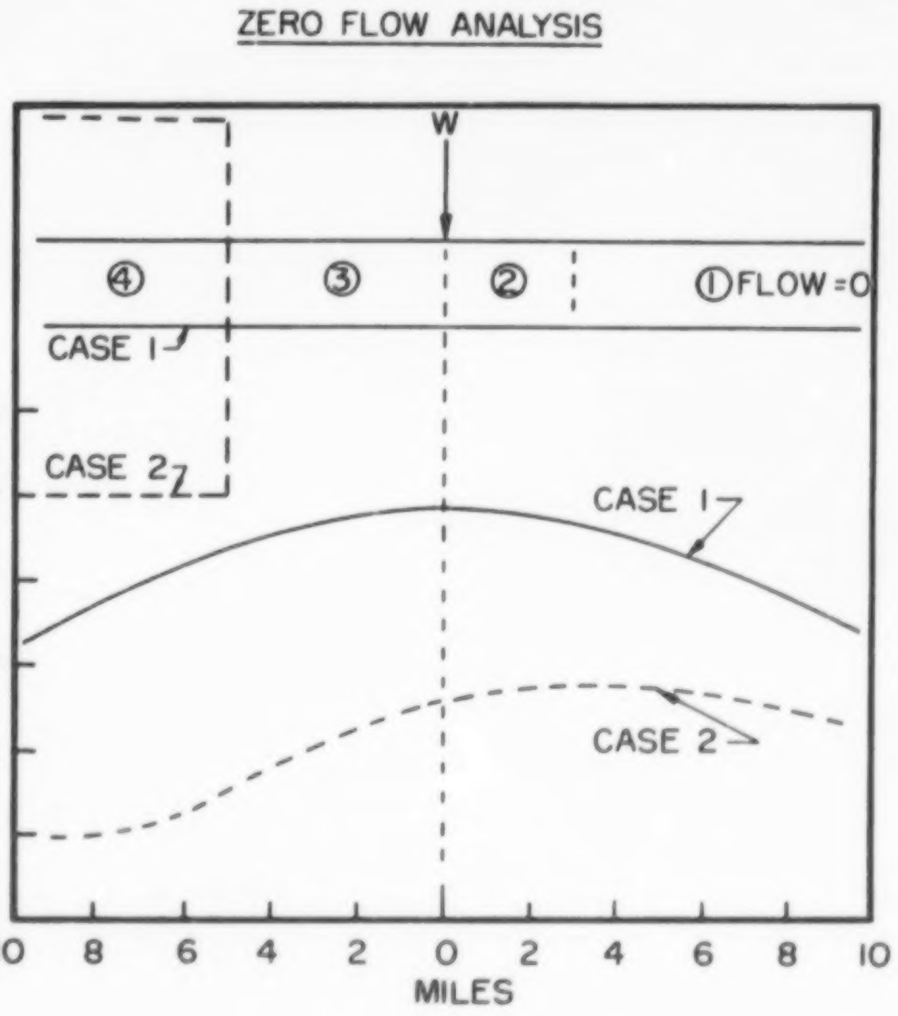

ZERO FLOW ANALYSIS
W
④
③
②
①FLOW=0
CASE 1
CASE 2
CASE 1
CASE 2
DISSOLVED OXYGEN DEFICIT (ppm)
0
1
2
3
4
5
6
10
8
6
4
2
0
2
4
6
8
10
MILES

FIGURE 18

ZERO FLOW ANALYSIS

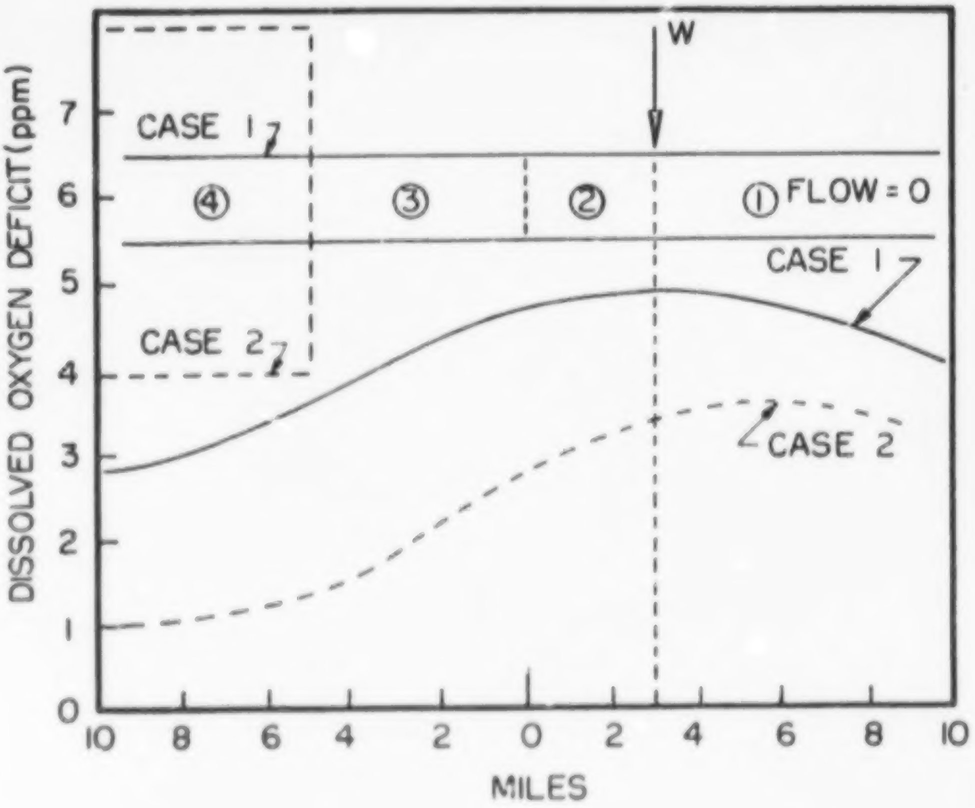


FIGURE 19

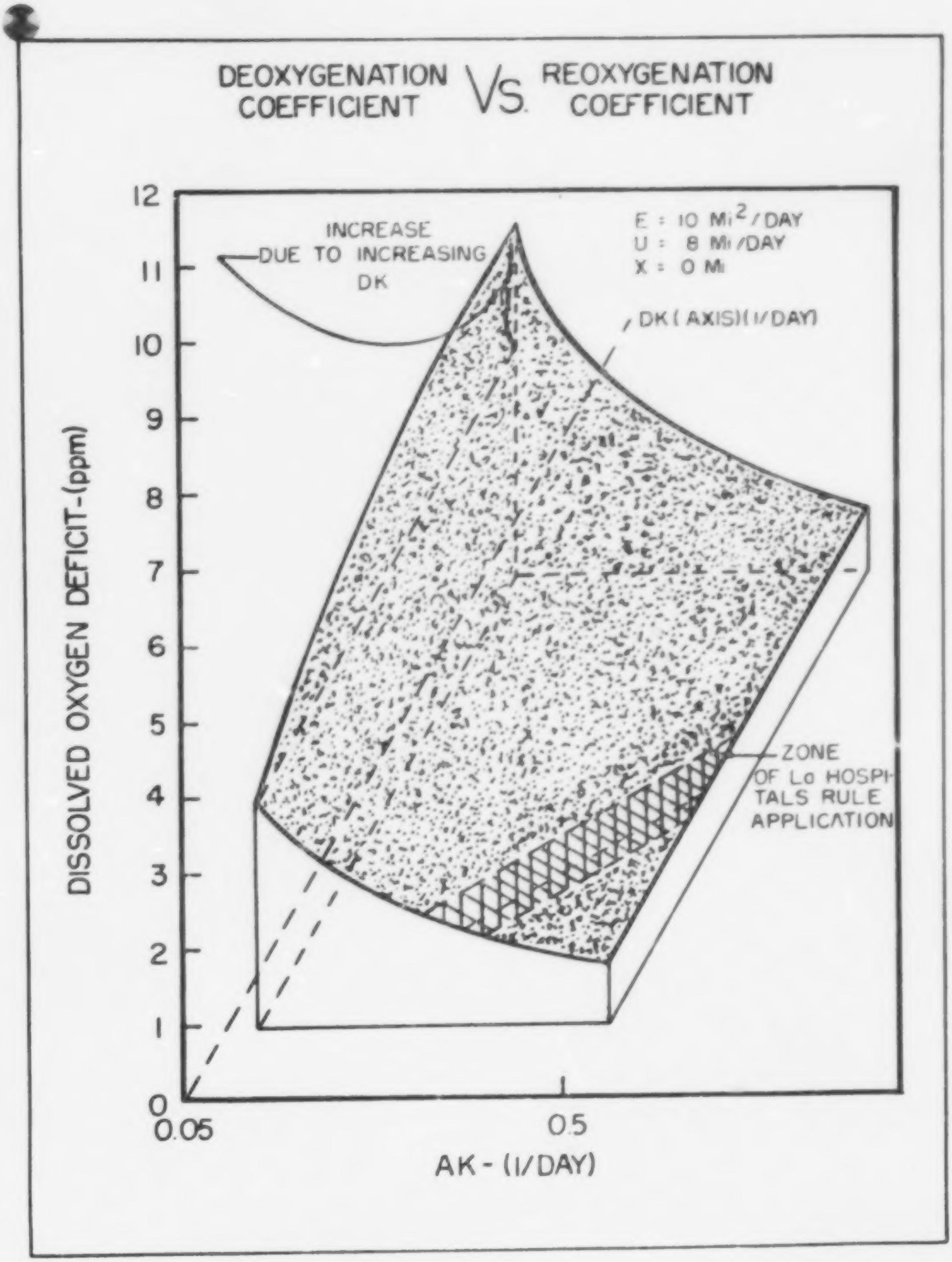

DEOXYGENATION COEFFICIENT VS. REOXYGENATION COEFFICIENT
E = 10 Mi^2/DAY
U = 8 Mi/DAY
X = 0 Mi
INCREASE DUE TO INCREASING DK
DK (AXIS)(1/DAY)
ZONE OF Lo HOSPITALS RULE APPLICATION
DISSOLVED OXYGEN DEFICIT-(ppm)
0
1
2
3
4
5
6
7
8
9
10
11
12
0.05
0.5
AK-(1/DAY)

FIGURE 20

DESCRIPTION OF THE COMPUTER PROGRAM

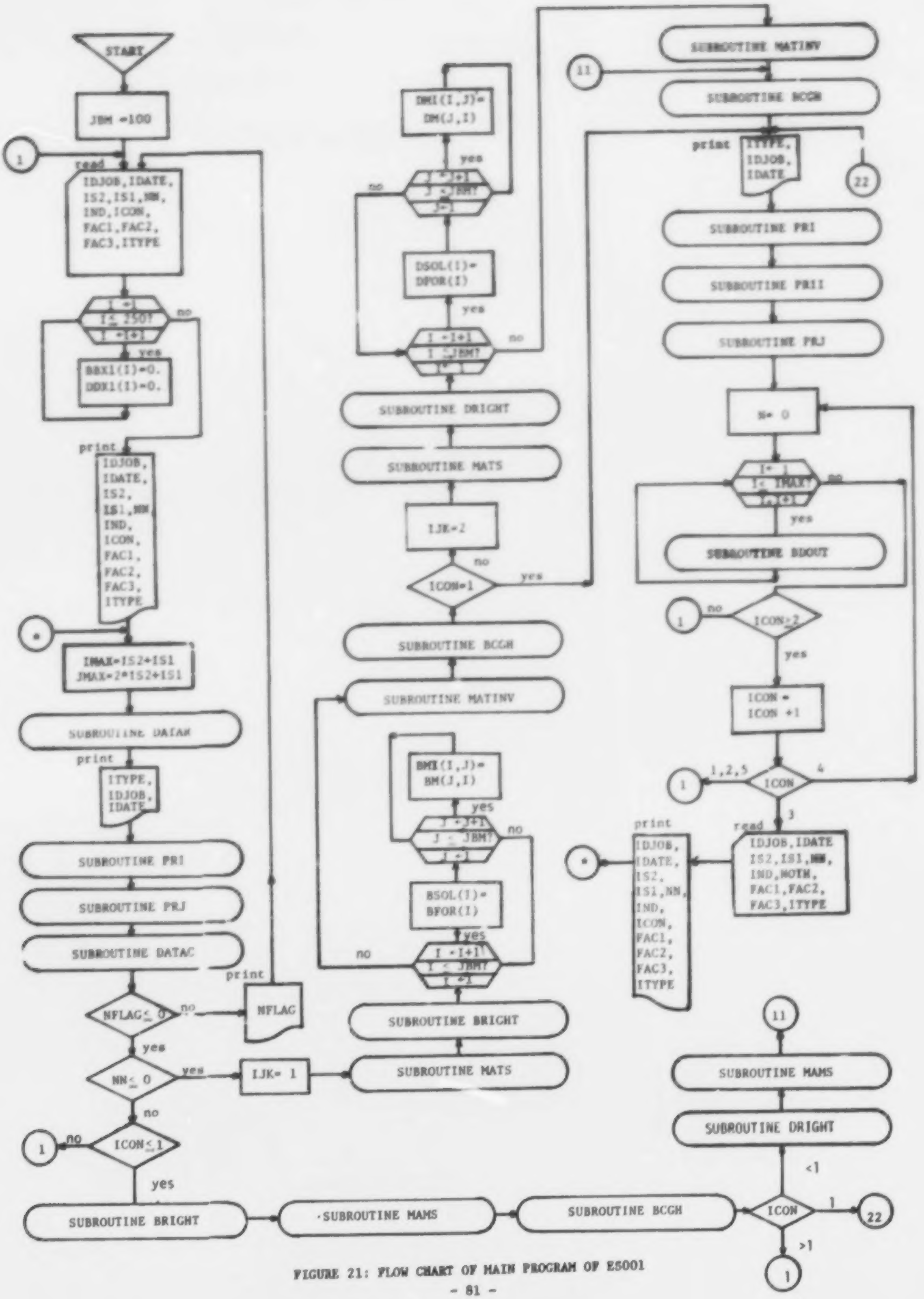


FIGURE 21: FLOW CHART OF MAIN PROGRAM OF ES001

The ES001 computer program was written for an IBM 370 machine in FORTRAN IV. In this section, the program will be described as follows:

1) a literal description of the main program accompanied by a flow diagram will be presented;

2) individual literal descriptions of each of the subroutines and functions used to support the main program will follow in alphabetical order;

MAIN PROGRAM

A flow chart of the main program is shown in Figure 21 and a listing in the appendix. As can be seen there are several branches due to a number of decision operations which are made in the course of ES001's execution. Most of these are due to the fact that the program has the capability of modelling several general systems, as well as that it has the option to test the response of a particular system to several sets of loads by saving the inverted matrix from the first run in a series. Since this is so, a straight description of the entire MAIN would be cumbersome and might be confusing. To alleviate this, the description will be presented in two parts. First, the program with the matrix computations is described, and the section of the flow chart dealing with it will be presented. Second, a run with no matrices computed will be explained and the section of the main flow chart associated with it will be presented.

[NOTE: A listing of ES002 and a discussion of that programs specifications is included in the appendix. ES002 is designed to run on an IBM 1130 System. For this reason, some of its code will differ from that of ES001 which is designed to run on an IBM 370 system. Therefore, the following sections will not necessarily be an accurate description of the ES002 algorithm. However, as a large part of the programs are similar this discussion may prove useful in working with ES002.

As noted previously, all input, output and internal computation for the programs are identical.]

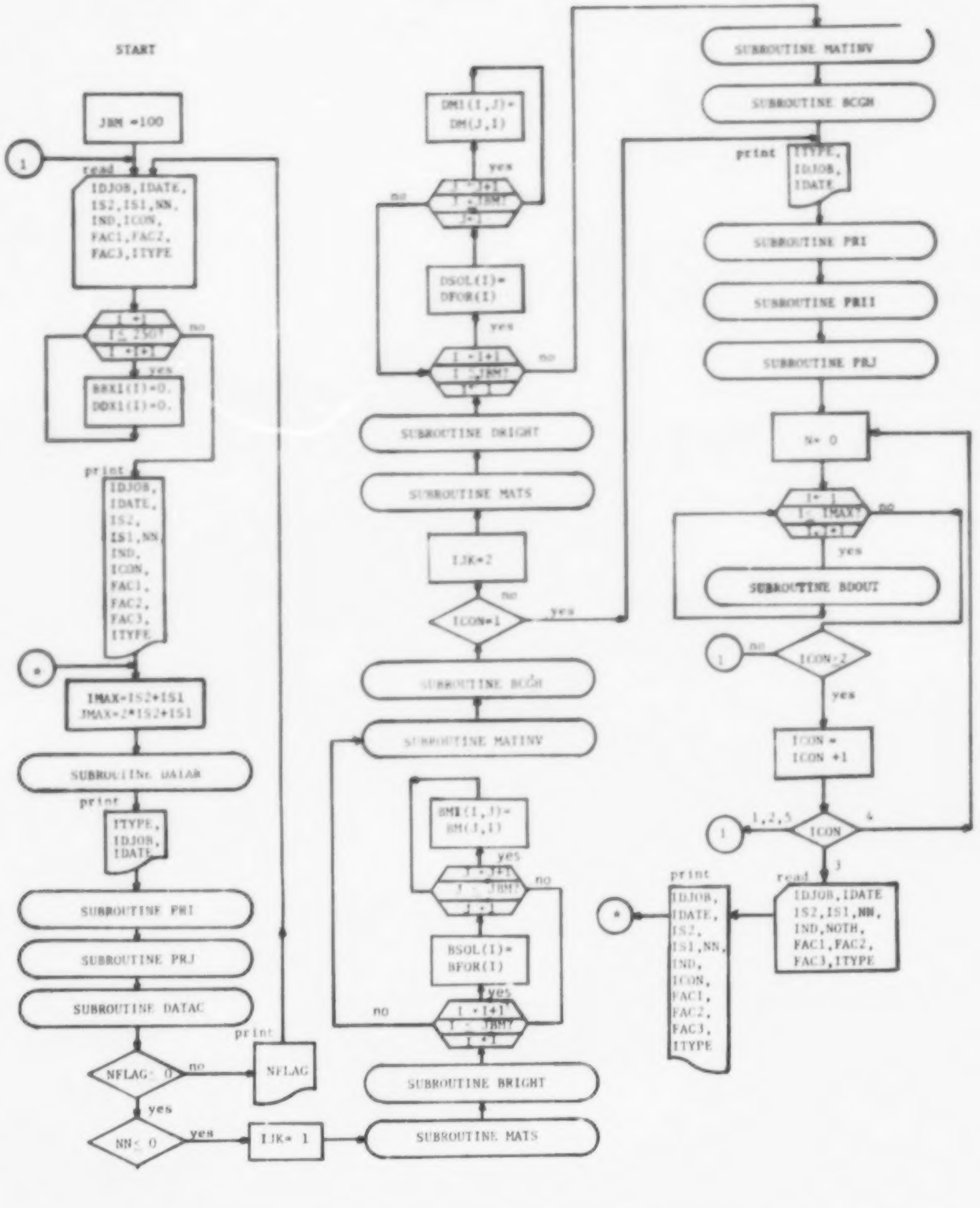


FIGURE 22: FLOW CHART OF MAIN PROGRAM RUN WITH MATRIX COMPUTATIONS

1) Run with matrix computations (flow chart on page 83).

The program starts by setting JBM, which is the maximum allowable number of boundary conditions, to 100. Then, the job identification card is read (see page 113 for description). After a do loop sets BBXI and DDXI to zero, the values that were read from the identification card are printed (as described on page 124). After IMAX (total number of section) and JMAX (total number of boundary conditions) are calculated, subroutine DATAR is called in order to read in and print out the remaining data cards (as they appear as card images) as well as to check the data for errors and print a number of diagnostics. If such errors exist a value of the parameter NFLAG >0 is returned to MAIN.

Subroutines PRI and PRJ, which list the data in tabular form, are called followed by DATAC which sets up temperature correction factors, conversion factors and does miscellaneous computations. At this point, if NFLAG is greater than zero the program essentially bombs out by returning to MAIN 014 where it reads the next set of data or ends if no data set is available.

If no errors were discovered (NFLAG = 0) the program proceeds to another decision where it is determined whether the model is a new one for which new matrices should be computed and inverted, or one for which the matrices have been previously computed but to which different loads are to be applied.

If the latter is true the program merely omits the steps in which the matrices are set up and inverted. (The actual mechanism is described in the next section.)

If the former is true and this is a new model (NN = 0), the program sets IJK = 1. This index connotes that the succeeding steps are for BOD. Subroutine MATS, which sets up the matrix (in this case for BOD) and subroutine BRIGHT which sets up the BOD forcing function are called followed by a do loop which redefines BFOR (BOD forcing function), which was output from BRIGHT, as BSOL. It also transposes the rows and columns of the matrix in preparation for the matrix inverse. MATINV then inverts the matrix and solves for the unknown coefficients. For output purposes BCGH then splits the coefficient matrix into B's and C's.

At this point, the BOD portion of the solution has been essentially computed. A decision is now made to determine whether a DO solution is desired. If it is, IJK is set to 2, MATS is called to set up the deficit matrix and DRIGHT generates the deficit forcing function. After a do loop transforms DFOR to DSOL and transposes the elements of the deficit matrix (as was done in the BOD), the matrix is inverted and BCGH is applied to the solution of coefficients. If a deficit solution had not been desired, the program would skip from MAIN 052 to this point, thus, omitting the deficit solution.

The particular model labels for this run are then printed and subroutine PRI, PRII, and PRJ print the input data in converted (or compatible) units as well as several computed parameters (e.g., velocity). After this N is set to zero and a do loop is used to print out the concentration profile for each section. At this point if ICON is less than 2 (that is, if we are not dealing with a coupled system with two inputs like carbonaceous and nitrogerous BOD), the program returns to MAIN 014 to begin another model run. If we are dealing with a coupled system with two inputs, ICON is incremented and a computed go to statement is employed to direct the flow (in this case ICON would now equal 3) to read and print the set of job identification cards for the second input and returns to MAIN 026 and computes and prints the BOD and DO solution as done previously and eventually returns to the decision at MAIN 070. There it would be decided that ICON would be incremented to 4. Then, using the computed go to of MAIN 072 the program flows to MAIN 067 where the sum of the profiles due to the two inputs is printed using BDOUT. ICON is finally incremented to 5 and the computed go to returns the flow to MAIN 014 where another run is started.

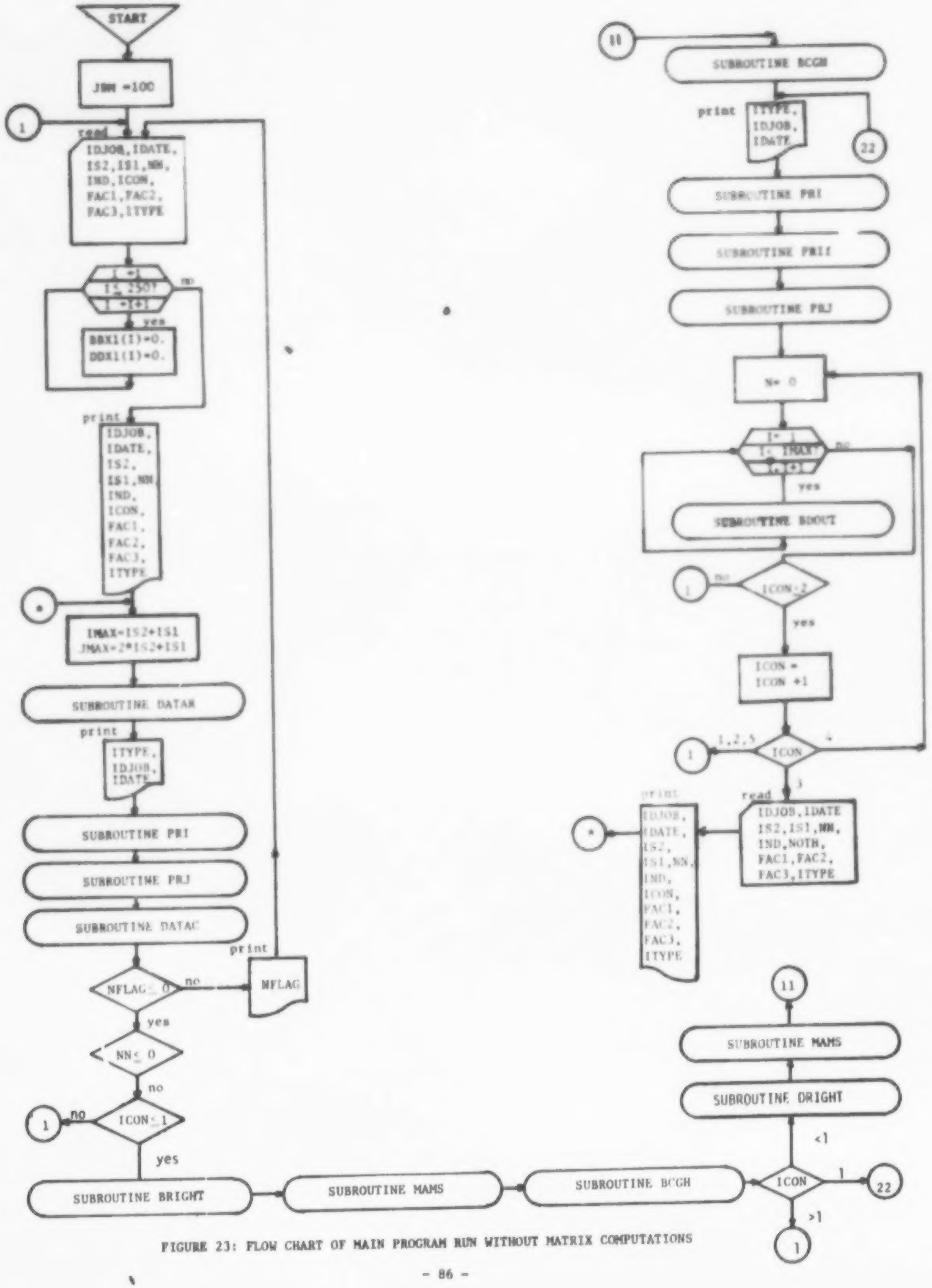


FIGURE 23: FLOW CHART OF MAIN PROGRAM RUN WITHOUT MATRIX COMPUTATIONS

2) Run without matrix computation (flow chart on page 86). This is the case where succeeding runs are being executed on the same system with different loads.

As can be seen from the flow chart the basic difference between this and the previous flow is that this run skips the matrix computations. This would occur in a case where several runs with different loads would be run in succession and recomputation of the matrix would be unnecessary. After the decision at MAIN 041, the program checks if ICON is greater than 1. If this is so, the run is terminated, since only one matrix can be saved by this program at a time (if it is dealing with a double input system it would only save the second matrix and erase the first). If ICON denotes that it is a single system or a coupled system of one input, the program proceeds to BRIGHT, MAMS AND BCGH which essentially set up and print the BOD matrix and forcing function. If the system is single (ICON = 1) the program proceeds to MAIN 063 where the data is printed in compatible units, the profile is outputted and the flow is returned to MAIN 014 where the next model is read. If the system is coupled (e.g., BOD and DO), DRIGHT, MAMS and BCGH are applied to generate and print the DO deficit matrix and forcing function. The data is then printed in compatible units, the profiles are output and the flow is returned to MAIN 014 where the next model is begun.

SUBROUTINES

Subroutine BCGH

MATINV outputs the solution for the unknown coefficients in the form of a single column of numbers. The purpose of BCGH is to transform this column into sets of coefficients corresponding to each section for output in PRI.

As was noted in the theory section the mathematical formulation for each double section generates two unknown coefficients while each single section generates one coefficient. Utilizing this fact, and the fact that the double sections have lower identification numbers according to the definitions of this program (see page 118), BCGH converts the single column of coefficients into pairs of subscripted B's and C's (for BOD) or G's and H's (for DO deficit) for each double section. After the double sections are completed, the subroutine continues to designate the coefficients according to the following standards:

	Coefficient	
KFO	BOD	DO Deficit
2 (- ∞)	B	G
3 (+ ∞)	C	H
4 (bay)	B	G

Subroutine BDOUT

BDOUT prints the concentration profile of the substance(s) being modelled for a particular section (page 127). Values are printed within the section for intervals of length PDEL or 1 mile if no PDEL is stipulated.

Use is made of BODO, a subroutine which actually computes the concentration for each point and stores values of BOD and DO for coupled systems with two components.

Subroutine BODO

Utilizing functions OM and YB, this subroutine generates and computes the final BOD and DO deficit equations (as on page 27).

The equations are calculated as follows:

1. BOD:

BOD = BBB = OM (KFO,B,C,FR,FBR,I,X) + YB (I,X)

where

OM - is a function described on page 101. In this particular case OM is the general solution to the BOD differential equation (equation 11, page 19).

YB - is a function described on page 102 which equals the particular solution to the BOD differential equation (equation 12a, page 21).

2. DO Deficit:

a) For double sections and infinity sections.

DO deficit = DDD = OM (KFO,G,H,FA,FBA,I,X) + ZZ

+ FAC(I) * OM (KFO,B,C,FR,FBR,I,X)

where

OM (KFO,G,H,FA,FBA,I,X) = In this case is the general solution to the DO deficit differential equation.

ZZ + FAC(I) * OM (KFO,B,C,FR,FBR,I,X) - This term is the particular solution to the DO deficit general solution. ZZ and FAC(I) were computed in DATAC (page 92).

b) For completely mixed bays.

DO deficit = DDD = OM (KFO,G,H,FA,FBA,I,X)

Subroutine BRIGHT

BRIGHT sets up the BOD forcing functions for each of the boundary conditions. Utilizing a computed GO TO, the subroutine sets up the following forcing functions for the appropriate boundary condition (as delineated in Table 6 on page 62):

Boundary Condition	Forcing Function
1, 2, 3	YB(K,Y) - YB(I,X)
4	- YB(I,X)
5, 6, 7	W(J) + Q(I) * YB(I,X) - Q(K) * YB(K,Y)
	W(J) + A(I) * YB(I,X) - Q(K) * YB(K,Y) - Q(KK) * YB(KK,Z)
9	W(J) + Q(I) * YB(I,X) + Q(KK) * YB(KK,Z) - Q(K) * YB(K,Y)
10	QD(J) * BDD(J) - Q(K) * YB(K,Y)
11	W(J) - Q(I) * YB(I,X)
12	W(J) - Q(I) * YB(I,X) - Q(K) * YB(K,Y)
13	W(J) + Q(I) * YB(K,Y)
14	W(J) + Q(I) * YB(K,Y) - Q(K) * YB(K,Y)

where:

W(J) = The point load at the junction in question.

Q(I,K or KK) = The advective flow in the subscripted section.

YB(I,K or KK; X, Y or Z) = This function is called to give the particular solution for the BOD equation as delineated in Table 1 on page 27. The algorithm of function YB is explained on page 104.

QD(J) = The advective flow over a dam.

BDD(J) = The BOD in the flow over a dam.

I,K or KK = Subscripts used to differentiate the sections which make up a junction.

Since the cards must always be inputted in order (either upstream or downstream) I usually connotes the first section, K the second and KK the third section forming a junction.

X, Y or Z = The distance of the junction from its reference junction for each segment making up the junction. X corresponds to segment I, Y to segment K, Z to segment KK. (They also correspond to XSA, XSB and XSC as noted on page 116 of the input description).

Finally, the forcing functions are printed.

Subroutine DATAC

This subroutine performs the following operations:

A. Temperature Correction Factors

AK(I) = AK(I) * FAC1 ** (TEMP(I) - 20.)

DK(I) = DK(I) * FAC2 ** (TEMP(I) - 20.)

RK(I) = RK(I) * FAC3 ** (TEMP(I) - 20.)

where

FAC1, FAC2 and FAC3 are temperature correction factors which were read in on the job identification card.

B. Conversion Factors to Compatible Units

1. The following conversion factors are defined:

a) CA = 1.0/(5280.0**2) - square feet to square miles

b) CV = 1./(5280.0**3) - cubic feet to cubic miles

c) CF = 3600.0*24.0/(5280.**3) - cfs to cubic mile/day

d) CL = (1.0E + 6)/(62.4*5280.**3)
-lbs/day to (mi^3/day) (mg/l) for point loads
-lbs/mile/day to (mi^2/day) (mg/l) for uniform loads

e) CM = 12./39.37 - feet to meters

2. These factors are then applied as follows:

a) AREA(I) = AREA(I) * CA

b) Q(I) = Q(I) * CF

c) WU(I) = WU(I) * CL

d) HT(I) = HT(I) * CM

e) VOL(I) = VOL(I) * CV

f) W(I) = W(I) * CL

g) QD(I) = QD(I) * CF

C. Miscellaneous variables are calculated:

a. FAC(I) = part of DO deficit forcing function =

DK(I) * FF(I)/(AK(I) - RK(I))

b. ZZ(I) = ((BD(I)/HT(I) + PR(I)

+ DK(I) * FF(I)/(RK(I) * AREA(I))/AK(I)

D. Forms and prints the symbolic BOD and DO deficit matrices using PLACE.

E. S, V, SD, VD are calculated and placed in common.

F. If present calculation is only for DO, sets BOD to 0.

Subroutine DATAR

This subroutine reads the data cards and prints them as described in the input description (pages 113 to 117). In the process of doing this, it also searches for input errors, prints an appropriate error message and aborts the run. Listed below are the errors and their associated messages:

Error	Message
Section #≤0*	Program aborts - no message printed
KFO is inconsistent with section type (double or single)*	Program aborts - no message printed
AK(I) = RK(I)	ERROR IN AK.OR.RK followed by the section number in which this occurred and the value of AK and RK.

---

*Uses NTPS.

Error	Message
HT = 0	Prints ERROR IN HT followed by the section number
Junction cards are incorrectly typed	Prints junction info followed by FAIL. If correct prints PASS.

Finally, the abbreviations for the 14 boundary conditions are assigned via a data statement (as explained on page 125).

Subroutine DRIGHT

DRIGHT sets up the dissolved oxygen deficit forcing functions for each of the boundary conditions. Utilizing a computed GO TO, the subroutine sets up the following forcing functions for the appropriate boundary condition (as delineated in Table 6, page 62):

Boundary Condition	Function
1, 2, 3	YD(K,Y) - YD(I,X)
4	- YD(I,X)
5, 6, 7	Q(K) * YD(I,X) - YDP(I,X) + YDP(K,Y) - Q(K) * YD(K,Y)
8	Q(I) * YD(I,X) - YDP(I,X) + YDP(K,Y) - Q(K) * YD(K,Y) + YDP(KK,Z) - Q(KK) * YD(KK,Z)
9	Q(I) * YD(I,X) - YDP(I,X) + YDP(K,Y) - Q(K) * YD(K,Y) + Q(KK) * YD(KKZ) - YDP(KK,Z)
10	YDP(K,Y) - Q(K) * YD(K,Y) + QD(J) * DD(J)
11	YDP(I,X) - Q(I) * YD(I,X) + FF(K) * DK(K) * B(K) * VOL(K)
12	YDP(I,X) - Q(K) * YD(I,X) + FF(KK) * DK(KK) * B(KK) * VOL(KK) + YDP(K,Y) - Q(I) * YD(K,Y)

Boundary Condition	Function
13	Q(I) * YD(I,X) - YDP(I,X) + FF(K) * DK(K) * B(K) * VOL(K)
14	Q(I) * YD(I,X) - YDP(I,X) + FF(KK) * DK(KK) * B(KK) * VOL(KK) + Q(K) * YD(K,Y) - YDP(K,Y)

where:

Q(I,K, or KK) = The advective flow in the subscripted section.

FF(I,K or KK) = The ratio of the ultimate to the 5-day BOD.

DK(K or KK) = Deoxygenation rate.

VOL (K or KK) = Volume of section.

OD(J) = Flow over a dam.

DD(J) = DO deficit over a dam.

B (K or KK) = Constant which was generated in the BOD solution.

YD(I,K, or KK; X,Y,Z) = Function which generates the particular solution to the DO deficit equation (as shown in Table 1, on page 27).

YDP(I,K or KK: X,Y or Z) = Function which generates the derivative of the particular solution to the DO deficit equation multiplied times the dispersion coefficient and the area.

Subroutine MAMS

This is only called if consecutive runs are being made with different loadings. In such a case, the forming and inversion of the matrices would be suppressed since these inverted matrices would already be in core. MAMS merely multiplies the inverted matrices by the new forcing functions which would be generated by the new loads.

Subroutine MATINV

This is a standard matrix inversion subroutine using an elimination method to invert the matrix and calculate the unknown coefficients utilizing the forcing function. If IND (printing indicator) is set to 2, it also prints out the original matrix and the forcing function and then the inverted matrix and the coefficients.

Subroutine MATS

This subroutine sets up the individual elements of the BOD or DO deficit matrix. It does this by evaluating each of the boundary conditions, using NOSEC to subscript the column location of the elements. In the case of double sections, two subscripts are allocated and in the case of single sections one subscript is designated. The following elements are computed for the associated boundary condition:

Boundary Condition	Matrix Elements
1	TM(IM,J) = F(I,X) TM(L,J) = FB(I,X) TM(KM,J) = -F(K,Y) TM(M,J) = -FB(K,Y)
2	TM(L,J) = F(I,X) TM(KM,J) = -F(K,Y) TM(M,J) = -FB(K,Y)
3	TM(IM,J) = F(I,X) TM(L,J) = FB(I,X) TM(M,J) = -FB(K,Y)
4	TM(IM,J) = F(I,X) TM(L,J) = FB(I,X) TM(M,J) = -1.0
5	TM(IM,J) = FP(I,X) - Q(I) * F(I,X) TM(L,J) = FBP(I,X) - Q(I) * FB(I,X) TM(KM,J) = Q(K) * F(K,Y) - FP(K,Y) TM(M,J) = Q(K) * FB(K,Y) - FBP(K,Y)
6	TM(L,J) = FP(I,X) - Q(I) * F(I,X) TM(KM,J) = Q(K) * F(K,Y) - FP(K,Y) TM(M,J) = Q(K) * FB(K,Y) - FBP(K,Y)
7	TM(IM,J) = FP(I,X) - Q(I) * F(I,X) TM(L,J) = FBP(I,X) - Q(I) * FB(L,X) TM(M,J) = Q(K) * FB(K,Y) - FBP(K,Y)

Boundary Condition	Matrix Elements
8	TM(IM,J) = FP(I,X) - Q(I) * F(I,X) TM(L,J) = FBP(I,X) - Q(I) * FB(I,X) TM(KM,J) = Q(K) * F(K,Y) - FP(K,Y) TM(M,J) = Q(K) * FB(K,Y) - FBP(K,Y) TM(N,J) = Q(KK) * FB(KK,Z) - FBP(KK,Z) TM(KKM,J) = Q(KK) * F(KK,Z) - FP(KK,Z)
9	TM(IM,J) = FP(I,X) - Q(I) * F(I,X) TM(L,J) = FBP(I,X) - Q(I) * FB(I,X) TM(KM,J) = Q(K) * F(K,Y) - FP(K,Y) TM(M,J) = Q(K) * FB(K,Y) - FBP(K,Y) TM(KKM,J) = FP(KK,Z) - Q(KK) * F(KK,Z) TM(N,J) = FBP(KK,Z) - Q(KK) * FB(KK,Z)
10	TM(KM,J) = Q(K) * F(K,Y) - FP(K,Y) TM(M,J) = Q(K) * FB(K,Y) - FBP(K,Y)
11	TM(IM,J) = Q(I) * F(I,X) - FP(I,X) TM(L,J) = Q(I) * FB(I,X) - FBP(I,X)
11 - BOD	TM(M,J) = VOL(K) * RK(K)
11 - DO deficit	TM(M,J) = VOL(K) * AK(K)
12	TM(IM,J) = Q(I) * F(I,X) - FP(I,X) TM(L,J) = Q(I) * FB(I,X) - FBP(I,X) TM(KM,J) = Q(K) * F(K,Y) - FP(K,Y) TM(M,J) = Q(K) * FB(K,Y) - FBP(K,Y)
12 - BOD	TM(N,J) = VOL(KK) * RK(KK)
12 - DO deficit	TM(N,J) = VOL(KK) * AK(KK)
13	TM(IM,J) = FP(I,X) - Q(I) * F(I,X) TM(L,J) = FBP(I,X) - Q(I) * FB(I,X)
13 - BOD	TM(M,J) = Q(K) + VOL(K) * (TIDE1(K) + FK(K))
13 - DO deficit	TM(M,J) = Q(K) + VOL(K) * (TIDE1(K) + AK(K))
14	TM(IM,J) = FP(I,X) - Q(I) * F(I,X) TM(L,J) = FBP(I,X) - Q(I) * FB(I,X) TM(KM,J) = FP(K,Y) - Q(K) * F(K,Y) TM(M,J) = FBP(K,Y) - Q(K) * FB(K,Y)

Boundary Condition	Matrix Elements
14 - BOD	TM(N,J) = Q(KK) + VOL(KK) * (TIDE1(KK) + RK(KK))
14 - DO deficit	TM(N,J) = Q(KK) + VOL(KK) * (TIDE1(KK) + AK(KK))

where

Q(I,K or KK) = The flow in a section.

VOL(K or KK) = The volume of a bay.

TIDE1(K or KK) = The tidal exchange coefficient of the section.

RK(K or KK) = BOD removal rate.

AK(K or KK) = Reaeration rate.

F, FB, FP, FBP are functions which are accessed via EXTERNAL and ENTRY statements from FUN and PFUN.

For BOD*

$F = e^{SX}$

$FB = e^{VX}$

$FP = Se^{SX}\ EA$

$FBP = Ve^{VX}\ EA$

For DO*

$F = e^{SDX}$

$FB = e^{VDX}$

$FP = SDe^{SDX}\ EA$

$FBP = VDe^{VDX}\ EA$

---

*See pages 50 & 54

Subroutine PLACE

This subroutine is called by DATAC to form the column location of the elements of the symbolic matrix which is printed as part of the output (p.126). The rows correspond to the particular boundary conditions while the columns are in terms of the coefficients to which the particular element applies (this is determined using NOSEC).

Subroutine PRI

This subroutine prints parameter values as described in the output description (p. 126). The data is printed in a more structured form than when originally printed via DATAR and headings are provided. It consists of three portions which are accessed by individual entry statements:

1) PRI - Prints the section data in either original or compatible units.

2) PRII - Prints section parameters which have been computed within program (e.g., S, V, U, ZZ, etc.).

3) PRJ - Prints the junction data in either original or compatible units.

Subroutine SVD

This subroutine calculates the coefficients (S, V, SD, VD) in the exponentials of the BOD and DO deficit solution of the differential equations.

a. For the BOD solution the equations are as follows:

(1) When the velocity $U = 0$.

$$S = RK/E \qquad V = RK/E$$

(2) When the velocity $U \neq 0$.

$$S = U/2E \left(1 + \sqrt{1 + \frac{4RK \cdot E}{U^2}}\right)$$

$$V = U/2E \left(1 - \sqrt{1 + \frac{4RK \cdot E}{U^2}}\right)$$

b. For the DO deficit solution the equations remain the same as above, except the quantities SD and VD are calculated and AK replaces RK.

FUNCTIONS

Function FUN

FUN utilizes entry statements to route various functions to PFUN as follows:

Entry	FUN
FR(I,X)	F(X,V,I,X)
FA(I,X)	F(SD,VD,I,X)
FBR(I,X)	FB(S,V,I,X)
FBA(I,X)	FB(SD,VD,I,X)
FPR(I,X)	FP(S,V,I,X)
FPA(I,X)	FP(SD,VD,I,X)
FBPR(I,X)	FBP(S,V,I,X)
FBPA(I,X)	FBP(SD,VD,I,S)
YBP(I,X)	YBPT(S,V,I,X)

The values of FUN are found in PFUN.

Function NOSECT

This is used to designate a subscript for each section so that any operation associated with that section will be consistently labelled.

If the section is a double section

$$\mathrm{NOSECT} = 2 * \mathrm{IL}$$

where

IL = The section number (ISA, ISB, ISC as defined on page 116 of the input description).

If the section is a single section

$$\mathrm{NOSECT} = 2 * \mathrm{IS2} + (\mathrm{IL} - \mathrm{IS2})$$

where

IS2 = The number of double sections.

Function NTPS

This function (called by subroutine DATAR) checks whether the sections are numbered correctly and whether the sections have a proper value of KFO.

It checks for the following convention, and if they are not met, it returns to DATAR with an appropriate value (>4) of NTPS which in turn causes the main program to terminate the present run:

1) Checks if the section number >0.

2) Section numbers for double sections ≤ IS2 (the total number of double sections).

3) KFO for double sections = 1.

4) Section number for single sections <JMAX = 2 * IS2 + IS1.

5) 1 < KFO (single section) ≤4 - if this is true it assigns NTPS - KFO.

Function OM

This function (called by subroutine BODO and functions YD and YDP) sets up the BOD and DO deficit solution and its derivative for use in YD and YDP. It uses KFO, which is an indicator of section type (either double (1), negative infinity (2), positive infinity (3), or completely mixed (4)), to direct a computed GO TO with the following results:

KFO	OM
1	BG(I) * F(I,X) + CH(I) * FB(I,X)
2	CH(I) * FB(I,X)
3	BG(I) * F(I,X)
4	BG(I)

where

BG(I) = B of the BOD solution or G or DO deficit.

CH(I) = C of the BOD solution of H of DO deficit.

F(I,X) and FB(I,X) = externalized function which through FUN and PFUN result in the appropriate exponential terms for the BOD or DO deficit solution or its derivative.

Function PFUN

This function subprogram calculates the following values which are accessed by entry statements:

Entry	PFUN
F(SSD,VVD,IX)	EXP(X * SSD(I))
FB(SSD,VVD,I,X)	EXP(X * VVD(I))
YBT(SSD,VVD,I,X)	WU(I)/(AREA(I) * RK(I))
FP(SSD,VVD,I,X)	SSD(I) * EXP(X * SSD(I)) * E(I) * AREA(I)
FBP(SSD,VVD,I,X)	VVD(I) * EXP(X * VVD(I)) * E(I) * AREA(I)
YBPT(SSD,VVD,I,X)	0

These functions are used throughout ES001 but particularly in MATS in calculating the elements of the matrices.

Function YB

YB = WU(I)/(AREA(I) * RK(I))

where

WU(I) = Uniform load to a section

AREA(I) = Cross-sectional area of the sectional

RK(I) = BOD removal rate of the section.

Function YD

YD calculates the particular solution to the DO deficit equations (as shown in Table 1 on page 27).

YD = FAC(I) * OM(KFO,B,C,FR,FBR,I,X) + ZZ(I)

where

FAC(I)* = DK(I) * FF(I)/(AK(I) - RK(I))

ZZ(I)* = ((BD(I)/HT(I)) + PR(I) + (DK(I) * FF(I)/(RK(I) * AREA(I))/AK(I)

OM(KFO,B,C,FR,FBR,I,X) = a function (page 101) which sets up the BOD solution via FUN and PFUN.

Function YDP

YDP sets up the first derivative of the particular solution to the DO deficit equations (as shown in Table 5 on page 54).

YDP = FAC(I) * OM(KFO,B,C,FPR,FBPR,I,X)

where

FAC(I) = Same as above in write-up of function YD.

OM(KFO,B,C,FPR,FBPR,I,X) = a function (page 101) which sets up the derivative of the BOD solution via FUN and PFUN.

---

*NOTE: FAC(I) and ZZ(I) were calculated in DATAC.

USER'S MANUAL

Initial Steps in Formulating ES001

The initial step in developing an analysis of water quality (in this case BOD and DO deficit in particular) is a critical review of the available information on the body of water under study. Specifically, the following physical, geophysical, and hydrological information must be obtained and evaluated.

1. The mean water depth of the various bodies of water should be determined.
2. The cross-sectional area of the various water bodies should be plotted as a function of location.
3. The drainage area should be evaluated as a function of distance, which provides an insight into fresh water flow quantities. As an example, if a small drainage area is associated with a section of estuary, it is unrealistic to anticipate large fresh water flows from run-off and normal ground water.
4. A qualitative evaluation should be made of the hydrograph of the system. This will assist in determining when field sampling programs might be carried out.
5. If the fresh water flow rate does not significantly change with distance, then the advective velocity of the system can be calculated. On the other hand, if area changes and/or fresh water flow inputs are such that the system advective velocity varies, plots of velocity with distance should be developed.
6. Knowing the tidal velocity in estuaries a plot of "AK", the reaeration coefficient vs. distance, using equation (49) to calculate "AK", is constructed.
7. The confluence of major water bodies should be noted on the distance plots.
8. The BOD removal and oxidation coefficients RK and DK should be evaluated from appropriate information.
9. The dispersion coefficient should be assigned, or evaluated.
10. Waste sources should be located and indications of the magnitude of loads obtained.

11. Chloride or other conservative substance profiles should be constructed. BOD and dissolved oxygen deficit profiles as a function of distance should be developed from any existing data.

It is usually advisable as a first step to construct a model incorporating as many simplifying assumptions as practical.

Having assessed the reliability of the available information on the physical hydrodynamic and input data, the various distance plots are compared in an attempt to minimize the number of segments required. Changes in area and flow may assumed to occur at the confluence of two bodies of water, thus reducing the number of segments required. Adjacent waste inputs can be combined at their center of gravity to reduce the number of segments. All of these steps are contingent upon the availability of data and the degree of complexity of the model justified by the available information (additional discussion of segmentation can be found on page 29).

Discussion of System Parameters:

It is possible to evaluate the various parameters, U, E, RK, DK, R, and AK from physical, hydrodynamic and observed data.

Advective Velocity (U)

The flow velocity which transports material out of the estuary can be evaluated:

$$U = \frac{Q}{A} \qquad (46)$$

Dispersion (E)

A plot of the log of the concentration of a conservative substance (usually chlorides) vs. distance results in a straight line. The slope of the line is equal to:

$$\text{Slope} = U/E$$

therefore;

$$E = U/\text{slope} \qquad (47)$$

BOD Removal and/or Deoxygenation (RK and/or DK)

A plot of the log of the BOD vs. distance results in a straight line. The slope of the line is defined by:

$$\text{Slope} = \frac{U}{2E} \left[1 \pm \left(1 + \frac{4 \cdot RK \cdot E}{U^2} \right)^{1/2} \right] \quad \ldots\ldots\ldots \quad (48)$$

The value of RK can be obtained from the solution of the above equation.

If the rate of BOD removal changes, the semi-log plot of BOD vs. distance will yield two (2) straight lines having different slopes. The slope of the line adjacent to the waste input will provide a measure of RK. The slope of the line farthest from the waste input is generally employed to calculate DK. The calculated value of DK must be equal to or less than RK. This analysis for calculating the values of RK and DK is valid only for BOD data which results from a single waste input. If the BOD profile is influenced by several waste inputs which cannot be grouped into a single source of BOD, the above analysis is not valid. A reasonable first assumption for the range of values of RK and DK is 0.20 - 0.50 to the base "e", with the two parameters assumed to be equal.

Reaeration Coefficient

This parameter can be calculated as follows:

$$AK = \frac{D \cdot U^{1/2}}{H^{3/2}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (49)$$

D = diffusivity in Equation 49

The velocity is the average velocity over a tidal cycle in estuaries.

Temperature Effects

Temperature influences the parameters AK, RK, and DK. Generally accepted temperature corrections for these parameters are:

$$AK_{(T)} = AK_{(20^\circ C)} \, 1.02^{(t - 20)} \quad \ldots\ldots\ldots\ldots\ldots \quad (50)$$

$$RK_{(T)} = RK_{(20°C)}\ 1.04^{(t - 20)} \ldots\ldots\ldots (51)$$

$$DK_{(T)} = DK_{(20°C)}\ 1.04^{(t - 20)} \ldots\ldots\ldots (52)$$

$$RK \quad = \text{Nitrogenous BOD } RK = RK_{N(20°C)}\ 1.08^{(t - 20)} \ldots\ldots (53)$$

In addition, temperature will also significantly influence the saturation value of dissolved oxygen. Tables are available indicating temperature and dissolved solids effects on the saturation value of oxygen in water.[9]

Tidal Exchange Coefficient

The tidal exchange, "R", provides a method of evaluating the transport of material out of a bay into an infinite medium like the ocean via the tidal volume (R is in units of 1/tidal cycle)

$$R = \frac{\text{Tidal Volume}}{\text{Mean Volume of the Bay}} \ldots\ldots\ldots (54)$$

Detailed discussion of the above parameters appears in several other sources.[2.5]

Program Run Preparation:

A. Prior to punching any data cards for the program run of the estuary that you are modelling, the following steps should be kept in mind.*

   1. Draw the symbolic river stretch (as in Figures 13 and 16 of the example problems).

   2. Decide where to section your model to consider waste loads (point, uniform, benthal, etc.), physical parameters and hydraulic parameters.

   3. Number your sections (See Note 1, p. 118).

   4. Determine your section types and define each junction as to the type of boundary equation which is to be applied there.

   5. Determine your reference junction(s) (Note 2, p. 118) and on your diagram place distances for each junction.

   6. List all required parameters for section and junctions in tabular form.

B. Punch the following input cards. (See Data Description.)

   1. Identification Card (1 card/run).

   2. Section Cards (A).

      a. One for each section and in consecutive ascending order.

---

*Reference will be made in this section to the example problems which begin on Page 130.

3. Section Cards (B).

    a. One for each section and in consecutive ascending order.

4. Junction Card-Concentration.

    a. One for each double junction, 2 for each triple junction, and none for a dam.

5. Junction Card-Mass Balance.

    a. One for each junction.

6. Repeat preceding 5 steps if you are using a model with 2 input components (i.e., carbonaceous and nitrogenous load).

C. Place cards in following order to run program (also see Figure 24).

1. Job Card.

2. Necessary JCL Cards.

3. Job ID Card.

4. Section Card A for 1st Section
" " B " " "
Section Card A for 2nd Section
" " B " " "
.
.
.
.
.
Section Card A for Nth Section
" " B " " "

5. Concentration Junction Card for Junction 1
Mass Balance Junction Card for Junction 1
Concentration Junction Card for Junction 2
Mass Balance Junction Card for Junction 2
.
.
.
.
.
Concentration Junction Card for Junction N
Mass Balance Junction Card for Junction N

6. /*

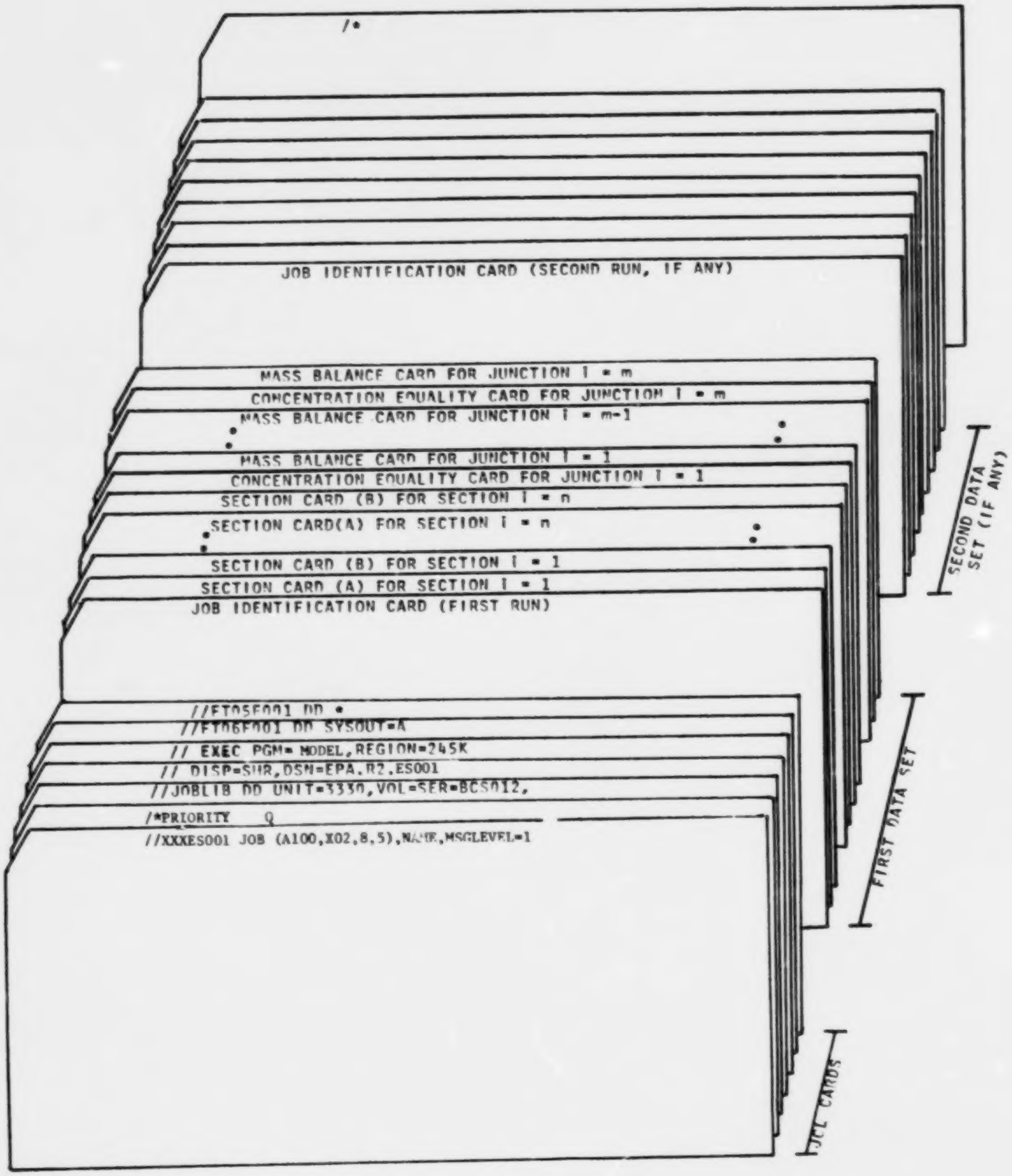
(NOTE: If there are more models or the run contains 2 input components, cards 3-5 from above are placed before the end of file (/*) card.

Figure 24

Deck set up for run using a load module on EPA computer (OSI)

n = total number of sections

m = total number of junctions


/*
JOB IDENTIFICATION CARD (SECOND RUN, IF ANY)
MASS BALANCE CARD FOR JUNCTION I = m
CONCENTRATION EQUALITY CARD FOR JUNCTION I = m
MASS BALANCE CARD FOR JUNCTION I = m-1
MASS BALANCE CARD FOR JUNCTION I = 1
CONCENTRATION EQUALITY CARD FOR JUNCTION I = 1
SECTION CARD (B) FOR SECTION I = n
SECTION CARD(A) FOR SECTION I = n
SECTION CARD (B) FOR SECTION I = 1
SECTION CARD (A) FOR SECTION I = 1
JOB IDENTIFICATION CARD (FIRST RUN)
SECOND DATA SET (IF ANY)
FIRST DATA SET
JCL CARDS

```
//FT05F001 DD *
//FT06F001 DD SYSOUT=A
// EXEC PGM= MODEL,REGION=245K
// DISP=SHR,DSN=EPA.R2.ES001
//JOBLIB DD UNIT=3330,VOL=SER=BCS012,
/*PRIORITY    Q
//XXXES001 JOB (A100,X02,8,5),NAME,MSGLEVEL=1
```

DATA DESCRIPTION & INPUT REQUIREMENTS (ALSO SEE DECK SET-UP):

Note: Superscripts refer to the notes following this section.

CARD 1 (JOB IDENTIFICATION AND DATE CARD)

Column(s)	Math Symbol	Variable	Description	Units	Format
1-4		IDJOB	Job name or no. (Arbitrary)		A4
5			blank		1X
6-9		IDATE	Date of run (month, year)		A4
10			blank		1X
11-13		IS2	No. of double sections		I3
14-16		IS1	No. of single sections		I3
17-19$^{(12)}$		NN	Indicator (0,1) which describes if: 0 - Current model is a new one. 1 - Current model is a repeat of the previous one but the waste load magnitude or its location have been changed.		I3
20-22		IND	Indicator (0,1) which causes the program to print (1) or not to print (0) the DO deficit and BOD matrices.		I3
23-25		ICON	Indicator (0,1,2) for coupled system, single system, and coupled with 2 inputs, respectively.		I3
26-30		FAC1	Temperature correction factor for AK.		F5.D
31-35		FAC2	Temperature correction factor for DK.		F5.D
36-40		FAC3	Temperature correction factor for RK.		F5.D
41-45			blank		5X
46-80		ITYPE	Description of system being modelled for identification purposes.		35A1

Card 2 (SECTION CARD A)

Column(s)	Math Symbol	Variable	Description	Units	Format
1		S	Symbol to identify section card) Used to help with deck set-up		1X
2		I	Designation of Section Card 1		1X
3-5		NS2	No. double sections (optional)		I3
6-8(1)		I	Section no. (in consecutive & ascending order)		I3
9-10			blank		
11-13		NJU(I)	Arbitrary name assigned to upstream junction of section		A3
14-18(2)		XJU(I)	Upstream distance of junction NJU(I) from reference junction	Mi.	F5.0
19-25	A	AREA(I)	Cross-sectional area of section	$Ft.^2$	F7.0
26-30(3)	RK	RK(I)	BOD removal coefficient of section	Day^{-1}	F5.0
31-35(3)	AK	AK(I)	Reaeration coefficient	Day^{-1}	F5.0
36-40	F	FF(I)	Ultimate to 5-day BOD ratio		F5.0
41-48	Q	Q(I)	*Flow	CFS	F8.0
49-56	BD	BD(I)	*Bottom oxygen demand (Benthal)	Gm/M^2-Day	F8.0
57-64	WU	WU(I)	*Uniform waste input	Lbs/Day-Mi.	F8.0
65-72(4)	R	TIDE1(I)	*Tidal exchange coefficient		F8.0
73-76			blank		
77-80		IDJ	Job identification (must agree with IDJOB-card 1)		A4

---

*Only required when available or applicable, otherwise blank or 0.

CARD 3 (SECTION CARD B)

Column(s)	Math Symbol	Variable	Description	Units	Format
1		S	Symbol to identify section card Used to help with deck set-up		1X
2		2	Description of section card 2		1X
3-5		NS1	No. single sections (optional)		I3
6-8(1)		I	Section no. (must agree with card A)		I3
9-10(5)		KFO(I)	Code no. for type of section 1 - Double section 2 - Negative infinity section 3 - Positive infinity section 4 - Completely mixed volume (BAY)		I2
11-13		NJD(I)	Arbitrary name assigned to downstream junction of section		A3
14-18(2)		XJD(I)	Downstream distance of junction NJD(I) from reference junction	Mi.	F5.0
19-25		Blank			7X
26-30	DK	DK(I)	BOD oxidation coefficient	Day^{-1}	F5.0
31-35	E	E(I)	Dispersion coefficient of section	$Mi.^2/Day$	F5.0
36-40	T	TEMP(I)	Temperature of section	°C	F5.0
41-48	R-P	PR(I)	*Algal photosynthesis and respiration	Mg/1-Day	F8.0
49-56	HT	HT(I)	Water depth	Ft.	F8.0
57-64	VOL	VOL(I)	*Volume of section (BAYS)	$Ft.^3$	F8.0
65-72(6)		PDEL(I)	Interval desired for output along section	Mi.	F8.0
73-76			blank		
77-80		ID(I)	Section ID used to identify section in output (arbitrary nomencl.)		A4

---

*Only required when available or applicable, otherwise blank or 0.

CARD 4 (JUNCTION CARD - CONCENTRATION)

Column(s)	Math Symbol	Variable	Description	Units	Format
1		J	Symbol to identify junction card (optional)		1X
2-3(7)		I	Junction card number		I2
4-6		JND(I)	Junction identification (same as NJD & NJU from section card)		A3
7-9(8)		JBV(I)	Boundary type of J'th junction, concentration		I3
10-12(9)		ISA(I)	'First' section forming junction		I3
13-18(2)		XSA(I)	Distance of 'first' sec. forming junction from its own reference junction	Mi.	F6.1
19-21(9)		ISB(I)	'Second' section forming junction		I3
22-27(2)		XSB(I)	Distance of 'second' section forming junction from its own reference junction	Mi.	F6.1
28-30(10)		ISC(I)	*'Third' section forming junction		I3
31-36(2)		XSC(I)	*Distance of 'third' section forming junction from its own reference junction (if triple junction)	Mi.	F6.1
37-42(11)	W	W(I)	*Point waste input	Lbs./Day	F6.0
43-48	QD	QD(I)	*Flow over dam	CFS	F6.0
49-54	DD	DD(I)	*Deficit at face of dam inflow OD	MG/LIT.	F6.0
55-60	LD	BDD(I)	*BOD in Flow OD	MG/LIT.	F6.0
61-76			blank		
77-80		IDJ	Job identification (same as card 1)		A4

---

*Only required when available or applicable, otherwise blank or 0.

CARD 5 (JUNCTION CARD - MASS BALANCE)

Column(s)	Math Symbol	Variable	Description	Units	Format
			SAME AS CARD 4 EXCEPT:		
2-3		I	Junction card number		I2
7-9$^{(8)}$		JBV(I)	Boundary type of J'th junction, mass balance		I3
28-30$^{(10)}$		ISC(I)	*'Third' section forming junction		I3
31-36$^{(2)}$		XSC(I)	*Distance of 'third' section of junction from its own reference junction (if triple junction)	Mi.	F6.1

---

*Only required when available or applicable, otherwise blank or 0.

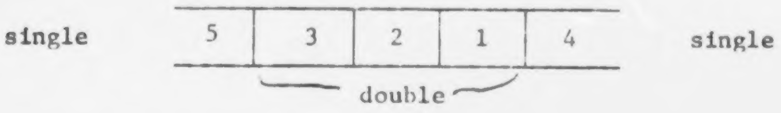
Notes on Input Cards

1. (Col. 6-8)

   (a) The consecutive order must be observed because any listing (by section) which is made by the computer program is in the form 1, 2, 3, . . . n (n = total number of sections) and it overrides your numbering system unless it is in the above form. For instance:

Your Input (If sections are numbered as follows)			Program Listing (Resulting Section Numbering)
1		2	1
2	or	4	2
3		6	3

   (b) In numbering your sections for a particular model the double sections must have a lower number than the single. If this is not observed the program prints out NCL (Non-Classifiable) and the program will not execute. For instance it must be:



2. (Col. 14-18)

   The reference junction is a junction that can be arbitrarily chosen to be the origin of your model and from which other sections can be located by a distance (miles). This distance can be positive or negative from the reference junction (where downstream is positive and upstream is negative).

   (a) The reference junction can be chosen at any point on your model.

   (b) There can be as many reference junctions as you have sections, however, when you choose such a junction you must decide which sections it is applicable to and any reference to these sections must be in terms of this reference junction.

(c) To simplify matters, however, choose as few reference junctions as possible and place them at strategic points.

(d) When deciding on your reference junction distance it is better to start with 0 mileage since this mileage is placed in an arithmetic expression of the form e^{kx} and if a combination of k and x becomes very large the solution might blow up.

(e) Junction AAA in example problem 1 and junction CCC in example problem 2 are reference junctions.

3. (Col. 26-36) and (Col. 31-35)
These two variables must not be equal since their difference appears in a denominator in the program. A warning is printed in regard to this and corrective action must be taken.

4. (Col. 65-72)

Only applicable to a completely mixed section where flow goes into the bay and is defined as:

$$\text{TIDE1} = \frac{\text{TIDAL VOLUME}}{\text{mean volume of bay.}} * \frac{\text{\# of tidal cycles}}{\text{day}}$$

Otherwise, equal to zero.

5. (Col. 9-10)

The negative infinity section is usually denoted by the uppermost upstream section and the positive infinity section as the farthest downstream section.

6. (Col. 65-72)

If PDEL is left blank the program sets its value to 1 mile.

7. (Col. 2-3)

Junction card numbers are in the form of 1, 2, 3, . . . n, (a) where n = total number of equations which is equal to: 2* number of double sections + number of single sections; (b) the junction card numbering must begin with 1 and proceed in consecutive ascending order, either upstream or downstream, to n. The order must be, a concentration junction card followed by a mass balance junction card or vice versa. This procedure is repeated as you move from junction to junction, with the number punched on each succeeding card increased by 1 until the n'th card.

8. The boundary type is a code number indicating the particular equation(s) to be applied at the junctions of interest depending on the types of sections contiguous to that junction. These boundary types and their numbers are as follows:

Concentration Equations

1 - Concentration equation for 2 double sections.

2 - Concentration equation for 1 double section and 1 negative infinity section.

3 - Concentration equation for 1 double section and 1 positive infinity section.

4 - Concentration equation for 1 double section and a completely mixed bay.

Mass Balance Equations

5 - Mass Balance equation for 2 double sections.

6 - Mass Balance equation for 1 double section and 1 negative infinity section.

7 - Mass Balance equation for 1 double section and 1 positive infinity section.

8 - Mass Balance equation for 3 double sections where flow from one divides into the other two.

9 - Mass Balance for 3 double sections where flow from 2 combines into the third.

10 - Mass Balance a section of advective flux only (dam).

11 - Mass Balance at a bay where flow leaves through the double section.

12 - Mass Balance at a bay where flow leaves through 2 double sections.

13 - Mass Balance at a bay where flow enters the bay from a double section.

14 - Mass Balance at a bay where flow enters the bay from 2 double sections.

9. (Col. 10-12) and (Col. 19-21)

In the variable description it states, first and second section forming a junction. This statement requires one to know not only the actual section numbers (which are read from your model diagram and are punched on the card) that form the junction but which of the two is the "first" (Col. 10-12) and which is the "second" (Col. 19-21). This first and second designation depends on section types (double, negative infinity, etc.) that meet at the junctions. Listed below are the 14 permissible section combinations and their card column positions for any one junction card. The list is in terms of symbols which are defined below:

▭ = double section, $-\infty$ = neg. inf., $+\infty$ = pos. inf., 0 = bay

Boundary Type (JBV)	First Section (ISA)	Second Section (ISB)	Third Section (ISC)
1	▭	▭	-
2	$-\infty$	▭	-
3	▭	$+\infty$	-
4	▭	0	-
5	▭	▭	-
6	$-\infty$	▭	-
7	▭	$+\infty$	-
8	▭	▭	▭
9	▭	▭	▭
10*	▭	▭	-
11	▭	0	-
12	▭	▭	0
13	▭	0	-
14	▭	▭	0

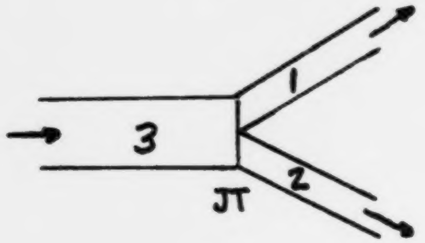
*Both section numbers are the same.

(NOTE: At a particular junction in any of the above cases where you have only 2 double sections, the upstream-downstream or downstream-upstream convention must be followed for the entire system being modeled. In other words, if you are numbering from upstream to downstream, at a particular junction of 2 double sections, the upstream section is the "first section" and the downstream section the "second section.").

10. (Col. 28-30)

When 3 double sections form a junction as in boundary conditions 8 and 9, the following placement of the sections must be adhered to on the Junction Cards.

(a) Flow from 1 Section into 2 Sections



For Junction JT:

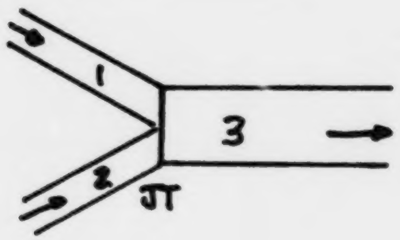
	ISA	ISB	ISC
2 Concentration Equality Cards	3 3	1 2	- -
1 Mass Balance Card	3	x*	x*

NOTE: Section 3 must be in the ISA location (Cols. 10-12).

---

*The x stands for either of the two other sections.

(b) Flow from 2 Sections into 1 Section



For Junction JT:

	ISA	ISB	ISC
2 Concentration	1	3	-
Equality Cards	2	3	-
1 Mass Balance Card	x*	3	x*

NOTE: Section 3 must be in the ISB location (Cols. 19-21).

11. (Col. 37-42)

Not allowed at a dam junction.

12. (Col. 17-19)

This value cannot be 1 if a coupled system with 2 input components is being modelled. The program will terminate if this is done.

---

*The x stands for either of the two other sections.

OUTPUT

1. INPUT DATA LISTING

The 80 column card images of the input data are read in and then printed in a format that spreads out the variables for easier reading.

a. The Job Identification and Date Card is listed first.

b. The section cards are listed next. As this is done, the word ERROR followed by the variable is printed when AK = RK* and/or HT = 0*. (See Input Restrictions, Notes B(1) and B(2)).

c. Immediately following the above, the sections are listed and identified by type according to the following:

(1) DBLE - A double section.

(2) S PI - A single positive infinity section.

(3) S NI - A single negative infinity section.

(4) S CM - A single completely mixed section (BAY).

In addition two error conditions are identified by MISS and NCL. Their meanings are as follows:

*(1) MISS - Type of section (KFO) is not punched on the section card.

*(2) N CL - Non-Classifiable. Either a section is not classified correctly according to (KFO) or a double or single section is not correctly classified according to Note 1 (p.118).

d. The junction cards are then listed and parentheses are added to more clearly identify the sections that meet at that junction and the mile point of that junction.

e. Immediately following the previous listing, a line is listed for each junction card. The following information is printed; equation number (which is identified in the input as junction card number), boundary type (JBV), abbreviation of the boundary type, and in parentheses the section numbers and type of sections (see (c) above) which meet at that junction.

---

*This condition will cause the run being done to be terminated.

The abbreviations of the 14 boundary types are as follows:

(1) C 2D - Concentration equation for 2 double sections

(2) C DN - Concentration equation for 1 double section and 1 negative infinity section

(3) C DP - Concentration equation for 1 double section and 1 positive infinity section

(4) C DC - Concentration equation for 1 double section and a completely mixed bay

(5) M 2D - Mass Balance equation for 2 double sections

(6) M DN - Mass Balance equation for 1 double section and 1 negative infinity section

(7) M DP - Mass Balance equation for 1 double section and 1 positive infinity section

(8) M3D1 - Mass Balance equation for 3 double sections where flow from one divides into the other two

(9) M3D2 - Mass Balance for 3 double sections where flow from 2 combines into the third

(10) M DM - Mass Balance a section of advective flux only (dam)

(11) MCM1 - Mass Balance at a bay where flow leaves through the double section

(12) MCM2 - Mass Balance at a bay where flow leaves through 2 double sections

(13) M1CM - Mass Balance at a bay where flow enters the bay from a double section

(14) M2CM - Mass Balance at a bay where flow enters the bay from 2 double sections

In addition at the end of each line the word PASS or FAIL* is printed. This denotes the correctness (PASS) or incorrectness (FAIL) of the punching of the section numbers on the junction cards according to the criteria of Note 9 (p. 121). The only exception being the case of only double sections appearing on a junction card. When this occurs the upstream-downstream rule of Note 9 must be followed and/or the special instructions for triple junctions (Note 10, p. 122).

---

*This condition will cause the run to be terminated.

2. TABLE OF SECTION AND JUNCTION PARAMETERS

a. The input data in the original units is listed in a more structured manner and with headings. The headings include the variable identification according to the Input Data Description, abbreviations of the variable names, and their units. The parameters pertaining to sections are listed first. This listing includes the section number, the section identification (ID), the type of section from 1(b) above, KFO, and the upstream and downstream junction identifications (NJU and NJD) with the actual mile point for each. This is followed by the data for each section (e.g., flow, area, etc.).

b. Immediately following are the junction card variables. Included on this Table are the equation numbers, the boundary type and the abbreviation for the boundary, the junction identification (JND), and in parentheses, the section numbers and the mile point of that specific junction, followed by the data.

3. MATRIX PRINT-OUTS

a. A symbolic matrix that is formed from the simultaneous linear equations is listed first. This matrix represents the positions of both BOD and D.O. deficit elements according to the inputting of the data cards and the segmentation of the estuary being modelled. The elements are in terms of the various junction identifications (JND's). There is also a row heading to denote the equation numbers and the boundary type (JBV). Each column corresponds to a particular coefficient of the solution.

*b. The Input BOD matrix is then printed. The elements of this are calculated from the input data and the BOD equations. The last row represents the elements in the BOD forcing function matrix.

*c. The inverted BOD matrix is then printed. At this point the units of the various elements are in DAY/MI^3. The last row now contains the solutions for the unknown BOD coefficients. (B and C for each section.)

*d. Immediately following the above the D.O. Deficit matrix is listed. The elements of this are calculated from the input data and the Deficit equations. The last row represents the elements in the Deficit force function matrix.

*e. Finally the inverted D.O. Deficit matrix is listed. As in the above the units of the various elements are in DAY/MI^3. The last row now contains the solutions for the unknown Deficit coefficients. (G and H for each section.)

---

*The printing of these matrices is optional and is controlled by IND of input card 1 (Job Identification and Date Card).

4. TABLE OF SECTION AND JUNCTION PARAMETERS IN COMPATIBLE UNITS

a. The next portion of the print-out lists the same output as in (2) above (structured form of the input data), but certain conversions have been applied to some of the variables to make the units compatible (see Subroutine DATA2).

b. In addition certain section variables, which have been computed in the program, are also listed. The additional variables and their definitions are as follows:

(1) U - The flow velocity in MI/DAY.

(2) S and V - The coefficients in the exponentials of the BOD solution.

(3) SD and VD - The coefficient in the exponential of the D.O. deficit solution.

(4) ZZ and FAC - Parts of the D.O. deficit forcing function.

$$ZZ = \frac{BD}{HT*AK} + \frac{PR}{AK} + \frac{DK*WU*FF}{AK*RK*A}$$

$$FAC = \frac{DK*FF}{AK-RK}$$

(5) The unknown section coefficients that have been solved for by the matrix inversion techniques.

(a) B and C - Values of the BOD coefficients in mg/l;

(b) G and H - Values of the D.O. deficit coefficients in mg/l.

*5. BOD AND D.O. DEFICIT SOLUTIONS

The resultant BOD and D.O. deficit values in mg/l are listed by ascending order of section and distance within sections according to the printing interval (PDEL) designated in the input cards. In addition, the section identification (ID) is printed on the left-hand side of the listing and the junction identification (JND) at both ends of the section is printed.

NOTE: If any negative BOD or D.O. deficit values are outputted or these values are not equal at section junctions, check the program restrictions and/or the input description of the variables.

---

*For a model that contains 2 waste input components such as carbonaceous and nitrogenous wastes, the data for each component will be printed along with its matrices and solutions as separate entities. The sum of their responses (as outlined in 5) will then follow.

RESTRICTIONS

NOTE: THE NUMBERS IN PARENTHESIS REFER TO THE NOTES ON DATA INPUT.

A. GENERAL

1. At the present there is a limit to the size of the estuary that can be modelled.

    a. In terms of junctions it means the maximum number of equations is 100.

    b. In terms of sections the following expression gives the limit. (2 x number of double sections + number of single sections) ≤ 100.

2. As with any model, ES001 has limits. When they are exceeded, ES001 will either reject the problem or compute erroneous values (e.g., negative values of BOD). The following guidelines are delineated to assist the programmer in formulating a realistic, executable model.

    If certain system parameters are exceeded the solution may blow up (approach infinite values or negative values). Fortunately, such extreme values are rarely encountered in actual estuaries. The following table gives suggested ranges of parameters which are within the limits of ES001.

Parameter	Acceptable Values
E	1-20 mi^2/day
Reaction rates	< 1/day
U	< 1 fps
Section length	< 20 miles

Table 7 Suggested parameter limitations for ES001.

These are by no means rigid constraints, as each particular configuration will have its exact limits, but care should be exercised when approach these extremities.

For instance, as the upper end of the estuary is approached, the dispersion coefficient (E) will decrease due to the absence of salinity gradients which are present in the downstream end. If E gets small enough it will cause the computation to blow up. This is due to the fact that the elements of the matrices are primarily of the form:

$$\frac{U}{2E}\ [1 \pm [1 + \frac{4K \cdot E}{U^2}]^{1/2}] \cdot X$$

where

X = mile point

U = velocity

K = reaction coefficient (AK in DO deficit solution or RK in BOD solution).

When E gets too small, the exponent of the above expression gets very large and in turn the entire expression gets very large. This will often cause the solution to blow up.

In such a case it would be advisable to shorten the section lengths and minimize the value of X by proper positioning of the reference junction. When a blowup occurs, the printout of the original matrix can often be useful in determining where the cause of the error is by examining each of the elements to see which are inordinately large.

B. SECTIONS

1. For any one section the reaeration (AK) coefficient cannot equal the removal (RK) coefficient(3).*
2. For any section that has a benthal waste source the depth (HT) cannot be zero(6).
3. For any run, the sections must be numbered consecutively with the double sections having lower numbers than the single(1).

C. JUNCTIONS

1. When punching the junction cards, the order must be upstream-down or vice versa(7).
2. The column location of the section numbers on any junction card has a specific order(9).
3. The program cannot handle anything greater than a triple junction and there are specific rules for a triple junction(10).

---

* Numbers in parenthesis refer to notes beginning on page 118.

EXAMPLE RUNS

A. The following two example RUNS will aid one in formulating a model to run in a manner that conforms with the computer program, and thus will result in the fewest errors when the first run is made.

To aid anyone with only slight familiarity with water quality models the following definitions are made:

a. Double Section (i.e., Sec. 1, Fig. 25) - A section that is bounded on both ends by another section.

b. Single Sections - End sections which are only bounded on one end by another section. The two types are:

(1) Infinity Sections - End sections where the BOD and DO deficits approach zero as the length of the section approaches infinity. When actually setting up a model this is usually limited to a few miles since this is the boundary of the system under consideration.

*(a) Negative Infinity (i.e., Sec. 2, Fig. 25) - An infinity section that is by convention the furthest upstream section.

*(b) Positive Infinity (i.e., Sec. 3, Fig. 25) - An infinity section that is by convention the furthest downstream section.

(2) Completely Mixed Volume (i.e., Sec. 7, 8, Fig. 28) - An example of this is a bay.

c Junctions

(1) Actual Junction - A river mile point where two or more sections meet or at which a dam is located.

(a) Single Junction - The point in a double section at which a dam is placed. (Fig. 28 .)

(b) Double Junction (i.e., JUNC BBB, Fig. 28) - A junction where two sections meet.

(c) Triple Junction (i.e., JUNC CCC, Fig. 28) - A junction where three sections meet.

---

*There can be more than one if the estuarine system you are modelling has tributaries entering it. The upstream and downstream on your diagram is shown by the direction of the flow (Q).

(2) End Section Junction (i.e., JUNC NBC, Fig.25) - The river mile points that define the limits of your model.

Now the following steps should be followed to prepare the 2 sample problems:

1. Draw the river stretch in a symbolic manner as in Figure 25 for the simple configuration, and Figure 28 for the more complex problem. (Figure 28 is derived from Figure 26.)

2. Section your model to consider

   a. Location of point, uniform and benthal sources and from Figure 28 the different types are:

      (1) Point Inputs - 5000 lbs./day at junction DDD and 1000 lbs./day in bay (Sec. 7) but since point waste loads can only be inputted via the junction cards, the load into the bay must be placed on the junction cards for junction GGG.

      (2) Uniform Waste Load - 500 lbs./day/mi. in Section 1.

   b. Physical parameters (i.e., cross-sectional areas).

   c. Hydraulic parameters (i.e., flow, dispersion).

   d. Other program characteristics as described in Restrictions (p.128).

3. Number your sections so that they are consecutive and the double sections have lower section numbers than the single.

4. Determine your reference junction(see page 118) (junction CCC on Fig. 28.) and place routine mileages on your diagram from this junction (mile 5 at junction DDD.)

5. Finally, give each junction an I.D. and determine the section types which are required on the input cards.

EXAMPLE RUN #1 - SIMPLE ESTUARY

A. INTRODUCTION

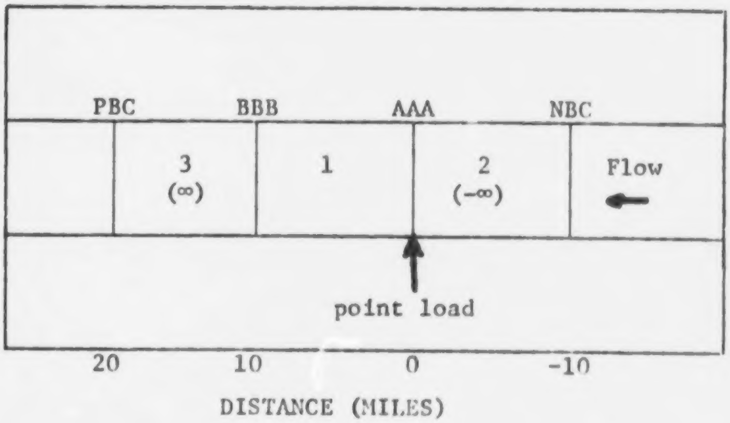


Figure 25 . A Simple Estuary.

Model the above estuarine system for dissolved oxygen and BOD using the ES001 program.

```
JOB ID AND DATE CARD
--------------------
SIMP     0672          1    2    0    0    0   1.020    1.040    1.040 EXAMPLE PROBLEM 1
INPUT SECTION CARD LISTING
--------------------------

   1   1   1   AAA    10.000   18800.0     0.300     0.264     1.250     375.0     0.0    0.0           0.0      SIMP

   2   1   1   BBB    20.000   18800.0     0.300     9.400    25.000       0.0     5.250  0.0           0.0      DOUB

   1   2   1   NBC     0.0     18800.0     0.300     0.264     1.250     375.0     0.0    0.0           0.0      SIMP

   2   2   2   AAA    10.000   18800.0     0.300     9.400    25.000       0.0     5.250  0.0           0.0      UXST

   1   3   1   BBB    20.000   18800.0     0.300     0.264     1.250     375.0     0.0    0.0           0.0      SIMP

   2   3   3   PBC    30.000      10.0     0.300     9.400    25.000       0.0     5.250  0.0           0.0      DNST

SECTION    1  15      DBLE

SECTION    2  15      5 NI

SECTION    3  15      S PI
INPUT JUNCTION CARD LISTING
---------------------------

   1 AAA    2       2   10.00       1   10.00      0    0.0   100000.0     0.0      0.0      0.0                SIMP

   2 AAA    6       2   10.00       1   10.00      0    0.0   100000.0     0.0      0.0      0.0                SIMP

   3 BBB    3       1   20.00       3   20.00      0    0.0        0.0     0.0      0.0      0.0                SIMP

   4 BBB    7       1   20.00       3   20.00      0    0.0        0.0     0.0      0.0      0.0                SIMP

EQUATION   1 OF TYPE  2     C DN   ( 2.S NI    )    ( 1.DBLE    )     ( 0.        )  PASS
EQUATION   2 OF TYPE  6     M DN   ( 2.S NI    )    ( 1.DBLE    )     ( 0.        )  PASS
EQUATION   3 OF TYPE  3     C DP   ( 1.DBLE    )    ( 3.S PI    )     ( 0.        )  PASS
EQUATION   4 OF TYPE  7     M DP   ( 1.DBLE    )    ( 3.S PI    )     ( 0.        )  PASS
```

```
                                          EXAMPLE PROBLEM 1
MODEL RUN SIMP      COMPUTED  0672      DATA GIVEN IN  ORIGINAL  UNITS


    SECTION      KFO  JUNCTION ID AND DISTANCE               RK        AK        DK          E          FF      TEMP   FDEL
                            AT ENDS                       REM.RATE  REA.RATE  DEO.RATE  DISP.RATE  ULT/5DAY         MILE-INCR
                                                          (1/DAY)   (1/DAY)   (1/DAY)   (MI2/DAY)           (CENT)   (MI)

   1 DOUB DBLE     1  (BBB,  20.0)   (AAA,  10.0)         0.300     0.264     0.300     9.400      1.250    25.000  0.0
   2 UKST S NI     2  (AAA,  10.0)   (NBC,   0.0)         0.300     0.264     0.300     9.400      1.250    25.000  0.0
   3 DNST S PI     3  (PBC,  30.0)   (BBB,  20.0)         0.300     0.264     0.300     9.400      1.250    25.000  0.0

    SECTION     CROSS          Q              BD              PR          TIDE1          HT       VOL           WU
               SEC-AREA       FLOW        BENTHL-DMD      PHOTO-RESP    TIDE-COEF       DEPTH    VOLUME      UNF-LOAD
               (FT**2)        (CFS)     (GMS/M2-DAY)    (MG/L-DAY)                    (FT.)    (FT**3)    (LBS/DAY-MI)
   1 DOUB DBLE  0.188E 05   0.375E 03       0.0             0.0           0.0         0.525E 01   0.0          0.0
   2 UKST S NI  0.188E 05   0.375E 03       0.0             0.0           0.0         0.525E 01   0.0          0.0
   3 DNST S PI  0.188E 05   0.375E 03       0.0             0.0           0.0         0.525E 01   0.0          0.0

     EQUATION     JUNCTION ID, SECTIONS                     W            QD          DD          BDD
                  MEETING AT JUNCTION,                   PT-LOAD      FLOW-DAM     DEF-DAM     BOD-DAM
                  AND DISTANCE                          (LBS/DAY)      (CFS)       (MG/L)      (MG/L)
   1   2 C DN AAA (  2 10.0)(  1 10.0)(  0  0.0)     100000.00000      0.0         0.0         0.0
   2   6 M DN AAA (  2 10.0)(  1 10.0)(  0  0.0)     100000.00000      0.0         0.0         0.0
   3   3 C DP BBB (  1 20.0)(  3 20.0)(  0  0.0)         0.0           0.0         0.0         0.0
   4   7 M DP BBB (  1 20.0)(  3 20.0)(  0  0.0)         0.0           0.0         0.0         0.0
```

```
ROW  JBV   1   2   3   4       SYMBOLIC DO DEFICIT AND BOD MATRIX

 1    2   AAA AAA AAA   -
 2    6   AAA AAA AAA   -
 3    3   BBB BBB   - BBB
 4    7   BBB BBB   - BBB
```

```
                                            EXAMPLE PROBLEM 1
MODEL RUN SIMP      COMPUTED  0672      DATA GIVEN IN COMPATIBLE UNITS


   SECTION      KFO  JUNCTION ID AND DISTANCE              RK        AK        DK          E          FF       TEMP   PDEL
                          AT ENDS                       REM.RATE  REA.RATE  DEC.RATE  DISP.RATE  ULT/5DAY          MILE-INCR
                                                        (1/DAY)   (1/DAY)   (1/DAY)   (MI2/DAY)            (CENT)  (MI)

 1 DOUB DBLE     1  (BBB.  20.0)   (AAA.  10.0)         0.365     0.291     0.365      9.400      1.250    25.000  0.0
 2 UXST S NI     2  (AAA.  10.0)   (NBC.   0.0)         0.365     0.291     0.365      9.400      1.250    25.000  0.0
 3 DNST S PI     3  (PBC.  30.0)   (BBB.  20.0)         0.365     0.291     0.365      9.400      1.250    25.000  0.0

   SECTION      CROSS        Q             BD              PR            TIDE1          HT        VOL          WU/AREA*RK
               SEC-AREA     FLOW       BENTHL-DMD      PHOTO-RESP      TIDE-COEF       DEPTH     VOLUME         UNF-LOAD
               (MI**2)   (MI**3/DAY)  (GMS/M2-DAY)    (MG/L-DAY)                      (MTRS)    (MI**3)          (MG/L)
 1 DOUB DBLE  0.674E-03  0.220E-03        0.0             0.0          0.0         0.160E 01  0.0              0.0
 2 UXST S NI  0.674E-03  0.220E-03        0.0             0.0          0.0         0.160E 01  0.0              0.0
 3 DNST S PI  0.674E-03  0.220E-03        0.0             0.0          0.0         0.160E 01  0.0              0.0

  SECTION        U          S         V          SD         VD          B          C          G          H          ZZ     FAC

 1 DOUB DBLE  0.32640   0.21518  -0.18045   0.19431  -0.15958  -0.00000  26.38181   0.00000 148.55461    0.0    -6.21
 2 UXST S NI  0.32640   0.21518  -0.18045   0.19431  -0.15958   0.50473   0.0        4.31480   0.0       0.0    -6.21
 3 DNST S PI  0.32640   0.21518  -0.18045   0.19431  -0.15958   0.0      26.38179   0.0      148.55455    0.0    -6.21

    EQUATION     JUNCTION ID. SECTIONS                               A              QD           DD          BDD
                 MEETING AT JUNCTION.                             PT-LOAD         FLOW-DAM     DEF-DAM      BOD-DAM
                 AND DISTANCE                            (MI**3/DAY)(MG/L)  (MI**3/DAY)   (MG/L)       (MG/L)
 1   2 C DN AAA (  2 10.0)(  1 10.0)(  0  0.0)              0.01089          0.0          0.0          0.0
 2   6 M DN AAA (  2 10.0)(  1 10.0)(  0  0.0)              0.01089          0.0          0.0          0.0
 3   3 C DP BBB (  1 20.0)(  3 20.0)(  0  0.0)              0.0              0.0          0.0          0.0
 4   7 M DP BBB (  1 20.0)(  3 20.0)(  0  0.0)              0.0              0.0          0.0          0.0
```

```
     SIMP                             EXAMPLE PPOBLEM 1
                                  COMPUTED     0672

   SECT.          SECT.          DIST.          BOD          D.O DEFICIT       JUNCTION
    NO.           NAME            MI            MG/L            MG/L              ID
   -----          -----          -----         ------        -----------       --------

    1             DOUB           10.0          4.3412          3.1775            AAA
    1             DOUB           11.0          3.6244          3.1832
    1             DOUB           12.0          3.0260          3.1097
    1             DOUB           13.0          2.5264          2.9816
    1             DOUB           14.0          2.1092          2.8179
    1             DOUB           15.0          1.7610          2.6327
    1             DOUB           16.0          1.4702          2.4368
    1             DOUB           17.0          1.2275          2.2380
    1             DOUB           18.0          1.0248          2.0420
    1             DOUB           19.0          0.8556          1.8528
    1             DOUB           20.0          0.7143          1.6730            BBB
```

```
                                      EXAMPLE PROBLEM 1
     SIMP                          COMPUTED      0672

SECT.          SECT.          DIST.          BOD          D.O DEFICIT          JUNCTION
 NO.           NAME            MI            MG/L            MG/L                 ID
-----          -----          -----         ------        -----------          --------

  2            UXST             0.0         0.5048          1.1822               NBC

  2            UXST             1.0         0.6260          1.3555

  2            UXST             2.0         0.7763          1.5467

  2            UXST             3.0         0.9626          1.7551

  2            UXST             4.0         1.1937          1.9785

  2            UXST             5.0         1.4803          2.2131

  2            UXST             6.0         1.8357          2.4525

  2            UXST             7.0         2.2764          2.6866

  2            UXST             8.0         2.8230          2.9010

  2            UXST             9.0         3.5007          3.0745

  2            UXST            10.0         4.3412          3.1775               AAA
```

```
                                   EXAMPLE PROBLEM 1
          SIMP                  COMPUTED     0672

 SECT.        SECT.         DIST.         BOD          D.O DEFICIT        JUNCTION
  NO.         NAME           MI           MG/L            MG/L               ID
 -----        -----         -----        ------        -----------        --------

   3          DNST          20.0         0.7143          1.6730             BBB

   3          DNST          21.0         0.5964          1.5043

   3          DNST          22.0         0.4979          1.3476

   3          DNST          23.0         0.4157          1.2032

   3          DNST          24.0         0.3471          1.0712

   3          DNST          25.0         0.2898          0.9511

   3          DNST          26.0         0.2419          0.8425

   3          DNST          27.0         0.2020          0.7446

   3          DNST          28.0         0.1686          0.6569

   3          DNST          29.0         0.1409          0.5784

   3          DNST          30.0         0.1175          0.5085             PBC
```

EXAMPLE RUN NUMBER 2

A. INTRODUCTION

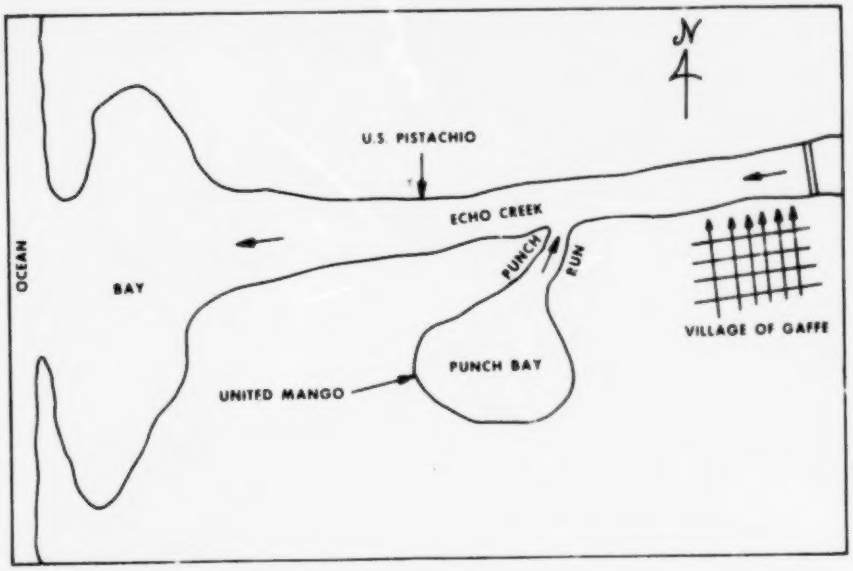


Figure 26 . ECHO CREEK.

Echo Creek is a fictitious polluted estuary in the Virgin Islands (see Fig. 26. Presently, two industries, United Mango, U.S. Pistachio, and one village, Gaffe, are located within the estuarine system. United Mango discharges its waste to Punch Bay which flows into Echo Creek via a one mile stretch called Punch Run. U.S. Pistachio discharges directly into the Echo at a point 5 miles downstream from its confluence with Punch Run. Gaffe, which is situated upstream from the confluence has no sanitary facilities and its waste eventually reaches the estuary in an untreated form. Gaffe's waste in contrast to the industries' contains an appreciable amount of settleable solids which have built up on the streambed as a sludge deposit. This deposit exerts a benthal demand on the dissolved oxygen concentration along the entire length of the estuary adjacent to the town. Finally, a dam is located at Gaffe's upstream boundary and a bay which is joined to the ocean is located at the downstream end of the Echo. The flow coming over the dam is slightly polluted.

A model is to be developed to determine the amount of waste treatment required by the municipality and the industries to improve water quality in the system to an acceptable level. Use of program ES001 will be employed to formulate such a model for BOD and DO deficit.

B. SEGMENTATION

Considering the confluence of Echo Creek and Punch Run as mile point zero, the following values for the cross-sectional area of the estuary are shown in Figure 27:

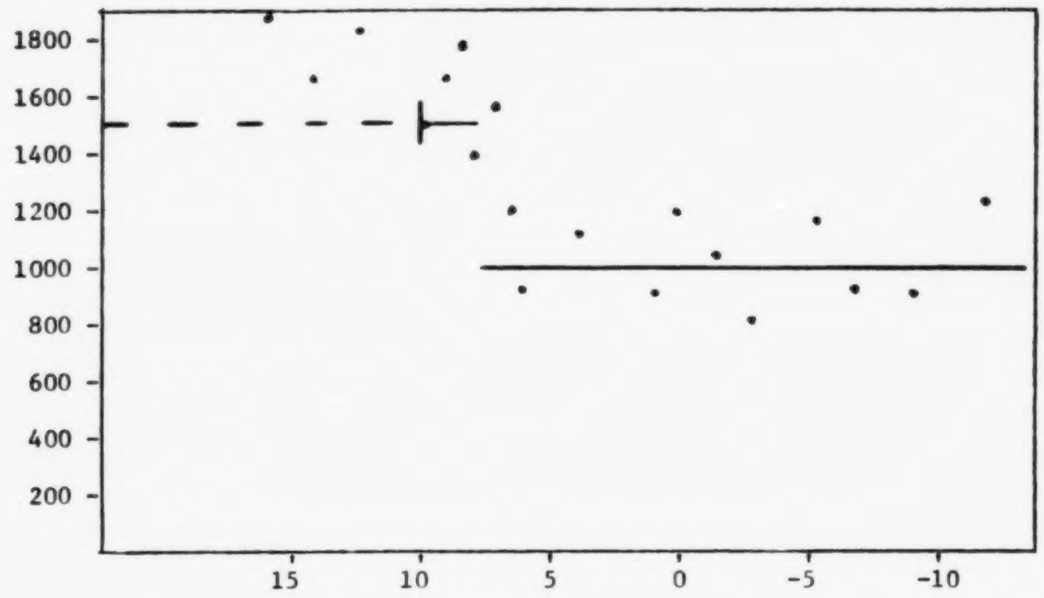


Figure 27. Plot of Cross-sectional Area Versus Distance Along Echo Creek.

With this in mind the following segments and junctions were determined:

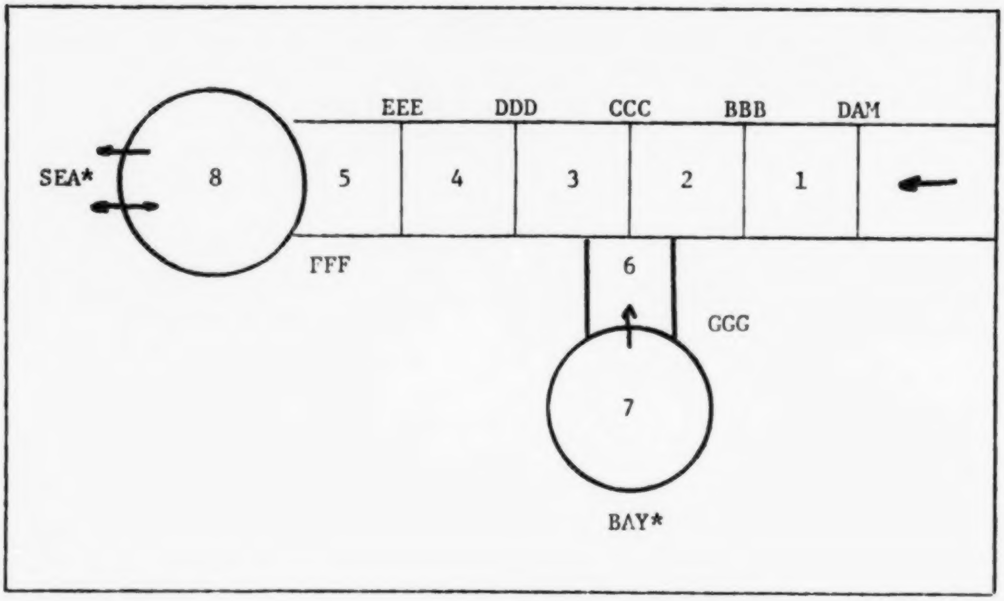


Figure 28. Segmentation of ECHO CREEK.

*NOTE: SEA and BAY are the names of pseudojunctions. They are meaningless but must be included when modelling bays.

```
JOB ID AND DATE CARD
--------------------
ECHO     0672        6    2    0    0    0   1.020    1.040    1.040 EXAMPLE PROBLEM 2
INPUT SECTION CARD LISTING
--------------------------

   6   1   1   DAM   -10.000    1000.0     0.275     0.100     1.250    100.0     2.000  0.100E 03     0.0      ECHO

   2   1   1   BBB    -5.000    1000.0     0.250     5.000    28.000      0.0    25.000  0.0           0.500    GAFF

   6   2   1   BBB    -5.000    1000.0     0.250     0.100     1.250    100.0     0.0    0.0           0.0      ECHO

   2   2   1   CCC     0.0      1000.0     0.250     5.000    28.000      0.0    25.000  0.0           1.000    ECUP

   6   3   1   CCC     0.0      1000.0     0.250     0.100     1.250    110.0     0.0    0.0           0.0      ECHO

   2   3   1   DDD     5.000    1000.0     0.250    10.000    28.000      0.0    25.000  0.0           1.000    ECJC

   6   4   1   DDD     5.000    1000.0     0.250     0.100     1.250    110.0     0.0    0.0           0.0      ECHO

   2   4   1   EEE     7.500    1000.0     0.250    10.000    28.000      0.0    25.000  0.0           1.000    ECDN

   6   5   1   EEE     7.500    1500.0     0.250     0.100     1.250    110.0     0.0    0.0           0.0      ECHO

   2   5   1   FFF    10.000    1500.0     0.250    15.000    28.000      0.0    30.000  0.0           0.0      USPS

   6   6   1   GGG    -1.000     100.0     0.250     0.200     1.250     10.0     0.0    0.0           0.0      ECHO

   2   6   1   CCC     0.0       100.0     0.250     5.000    30.000      0.0    10.000  0.0           0.250    PUNC

   6   7   1   BAY   -10.000     100.0     0.250     0.050     1.250    100.0     0.0    0.0           0.0      ECHO

   2   7   4   GGG    -1.000     100.0     0.250     5.000    30.000      0.0    40.000  0.400E 06     1.000    PBAY

   6   8   1   FFF    10.000     100.0     0.250     0.100     1.250    110.0     0.0    0.0           0.500    ECHO

   2   8   4   SEA    15.000     100.0     0.250     5.000    28.000      0.0    40.000  0.120E 09     5.000    ECBY

SECTION    1  1S      DBLE

SECTION    2  1S      DBLE

SECTION    3  1S      DBLE
SECTION    4  1S      DBLE

SECTION    5  1S      DBLE

SECTION    6  1S      DBLE

SECTION    7  1S      S CM

SECTION    8  1S      S CM
INPUT JUNCTION CARD LISTING
---------------------------

   1 DAM   10     1  -10.00     1  -10.00      0   0.0       0.0  100.0    0.2    0.2              ECHO

   2 BBB    1     1   -5.00     2   -5.00      0   0.0       0.0    0.0    0.0    0.0              ECHO

   3 BBB    5     1   -5.00     2   -5.00      0   0.0       0.0    0.0    0.0    0.0              ECHO

   4 CCC    1     2    0.0      3    0.0       0   0.0       0.0    0.0    0.0    0.0              ECHO

   5 CCC    1     4    0.0      3    0.0       0   0.0       0.0    0.0    0.0    0.0              ECHO

   6 CCC    9     2    0.0      3    0.0       6   0.0       0.0    0.0    0.0    0.0              ECHO

   7 DDD    1     3    5.00     4    5.00      0   0.0    5000.0    0.0    0.0    0.0              ECHO

   8 DDD    5     3    5.00     4    5.00      0   0.0    5000.0    0.0    0.0    0.0              ECHO

   9 EEE    1     4    7.50     5    7.50      0   0.0       0.0    0.0    0.0    0.0              ECHO

  10 EEE    5     4    7.50     5    7.50      0   0.0       0.0    0.0    0.0    0.0              ECHO

  11 FFF    4     5   10.00     8   10.00      0   0.0       0.0    0.0    0.0    0.0              ECHO

  12 FFF   13     5   10.00     8   10.00      0   0.0       0.0    0.0    0.0    0.0              ECHO

  13 GGG    4     6   -1.00     7   -5.00      0   0.0    1000.0    0.0    0.0    0.0              ECHO

  14 GGG   11     6   -1.00     7   -5.00      0   0.0    1000.0    0.0    0.0    0.0              ECHO
```

```
EQUATION  1 OF TYPE 10   M DM  ( 1.DBLE   )   ( 1.DBLE   )   ( 0.       )  PASS
EQUATION  2 OF TYPE  1   C 2D  ( 1.DBLE   )   ( 2.DBLE   )   ( 0.       )  PASS,
EQUATION  3 OF TYPE  5   M 2D  ( 1.DBLE   )   ( 2.DBLE   )   ( 0.       )  PASS
EQUATION  4 OF TYPE  1   C 2D  ( 2.DBLE   )   ( 3.DBLE   )   ( 0.       )  PASS
EQUATION  5 OF TYPE  1   C 2D  ( 6.DBLE   )   ( 3.DBLE   )   ( 0.       )  PASS
EQUATION  6 OF TYPE  9   M3D2  ( 2.DBLE   )   ( 3.DBLE   )   ( 6.DBLE   )  PASS
EQUATION  7 OF TYPE  1   C 2D  ( 3.DBLE   )   ( 4.DBLE   )   ( 0.       )  PASS
EQUATION  8 OF TYPE  5   M 2D  ( 3.DBLE   )   ( 4.DBLE   )   ( 0.       )  PASS
EQUATION  9 OF TYPE  1   C 2D  ( 4.DBLE   )   ( 5.DBLE   )   ( 0.       )  PASS
EQUATION 10 OF TYPE  5   M 2D  ( 4.DBLE   )   ( 5.DBLE   )   ( 0.       )  PASS
EQUATION 11 OF TYPE  4   C DC  ( 5.DBLE   )   ( 8.S CM   )   ( 0.       )  PASS
EQUATION 12 OF TYPE 13   M1CM  ( 5.DBLE   )   ( 8.S CM   )   ( 0.       )  PASS
EQUATION 13 OF TYPE  4   C DC  ( 6.DBLE   )   ( 7.S CM   )   ( 0.       )  PASS
EQUATION 14 OF TYPE 11   MCM1  ( 6.DBLE   )   ( 7.S CM   )   ( 0.       )  PASS
```

```
                                              EXAMPLE PROBLEM 2
MODEL RUN ECHO      COMPUTED  0672      DATA GIVEN IN  ORIGINAL  UNITS


   SECTION      KFO  JUNCTION ID AND DISTANCE          RK        AK        DK         E         FF       TEMP   PDEL
                          AT ENDS                   REM.RATE  REA.RATE  DEC.RATE  DISP.RATE  ULT/5DAY         MILE-INCR
                                                    (1/DAY)   (1/DAY)   (1/DAY)   (MI2/DAY)           (CENT)  (MI)

 1 GAFF DBLE      1  (BBB.  -5.0)  (DAM.  -10.0)    0.275     0.100     0.250      5.000     1.250    28.000  0.500
 2 ECUP DBLE      1  (CCC.   0.0)  (BBB.   -5.0)    0.250     0.100     0.250      5.000     1.250    28.000  1.000
 3 ECJC DBLE      1  (DDD.   5.0)  (CCC.    0.0)    0.250     0.100     0.250     10.000     1.250    28.000  1.000
 4 ECDN DBLE      1  (EEE.   7.5)  (DDD.    5.0)    0.250     0.100     0.250     10.000     1.250    28.000  1.000
 5 USPS DBLE      1  (FFF.  10.0)  (EEE.    7.5)    0.250     0.100     0.250     15.000     1.250    28.000  0.0
 6 PUNC DBLE      1  (CCC.   0.0)  (GGG.   -1.0)    0.250     0.200     0.250      5.000     1.250    30.000  0.250
 7 PBAY S CM      4  (GGG.  -1.0)  (BAY.  -10.0)    0.250     0.050     0.250      5.000     1.250    30.000  1.000
 8 ECBY S CM      4  (SEA.  15.0)  (FFF.   10.0)    0.250     0.100     0.250      5.000     1.250    28.000  5.000


   SECTION       CROSS         Q            BD           PR          TIDE1           HT         VOL           WU
                SEC-AREA      FLOW      BENTHL-DMD   PHOTO-RESP    TIDE-COEF        DEPTH      VOLUME       UNF-LOAD
                (FT**2)       (CFS)    (GMS/M2-DAY)  (MG/L-DAY)                     (FT.)      (FT**3)    (LBS/DAY-MI)

 1 CAFF DBLE   0.100E 04   0.100E 03     0.200E 01      0.0          0.0          0.250E 02   0.0          0.100E 03
 2 ECUP DBLE   0.100E 04   0.100E 03     0.0            0.0          0.0          0.250E 02   0.0          0.0
 3 ECJC DBLE   0.100E 04   0.110E 03     0.0            0.0          0.0          0.250E 02   0.0          0.0
 4 ECDN DBLE   0.100E 04   0.110E 03     0.0            0.0          0.0          0.250E 02   0.0          0.0
 5 USPS DBLE   0.150E 04   0.110E 03     0.0            0.0          0.0          0.300E 02   0.0          0.0
 6 PUNC DBLE   0.100E 03   0.100E 02     0.0            0.0          0.0          0.100E 02   0.0          0.0
 7 PBAY S CM   0.100E 03   0.100E 03     0.0            0.0          0.0          0.400E 02   0.400E 06    0.0
 8 ECBY S CM   0.100E 03   0.110E 03     0.0            0.0          0.500E 00    0.400E 02   0.120E 09    0.0

    EQUATION     JUNCTION ID. SECTIONS                    W            QD          DD          BDD
                 MEETING AT JUNCTION.                  PT-LOAD      FLOW-DAM     DEF-DAM     BOD-DAM
                 AND DISTANCE                         (LBS/DAY)      (CFS)       (MG/L)      (MG/L)

 1 10 M DM DAM (  1-10.0)(  1-10.0)(  0  0.0)          0.0        100.00000     0.20000     0.20000
 2  1 C 2D BBB (  1 -5.0)(  2 -5.0)(  0  0.0)          0.0          0.0         0.0         0.0
 3  5 M 2D BBB (  1 -5.0)(  2 -5.0)(  0  0.0)          0.0          0.0         0.0         0.0
 4  1 C 2D CCC (  2  0.0)(  3  0.0)(  0  0.0)          0.0          0.0         0.0         0.0
 5  1 C 2D CCC (  6  0.01(  3  0.01(  0  0.01        0.0           0.0      0.0      0.0
 6  9 M3D2 CCC (  2  0.01(  3  0.01(  6  0.01        0.0           0.0      0.0      0.0
 7  1 C 2D DDD (  3  5.01(  4  5.01(  0  0.01     5000.00000       0.0      0.0      0.0
 8  5 M 2D DDD (  3  5.01(  4  5.01(  0  0.01     5000.00000       0.0      0.0      0.0
 9  1 C 2D EEE (  4  7.51(  5  7.51(  0  0.01        0.0           0.0      0.0      0.0
10  5 M 2D EEE (  4  7.51(  5  7.51(  0  0.01        0.0           0.0      0.0      0.0
11  4 C DC FFF (  5 10.01(  8 10.01(  0  0.01        0.0           0.0      0.0      0.0
12 13 M1CM FFF (  5 10.01(  8 10.01(  0  0.01        0.0           0.0      0.0      0.0
13  4 C DC GGG (  6 -1.01(  7 -5.01(  0  0.01     1000.00000       0.0      0.0      0.0
14 11 MCM1 GGG (  6 -1.01(  7 -5.01(  0  0.01     1000.00000       0.0      0.0      0.0
                                SYMBOLIC DO DEFICIT AND BOD MATRIX
ROW  JBV   1   2   3   4   5   6   7   8   9  10  11  12  13  14

 1   10  DAM DAM   -   -   -   -   -   -   -   -   -   -   -   -

 2    1  BBB BBB BBB BBB   -   -   -   -   -   -   -   -   -   -

 3    5  BBB BBB BBB BBB   -   -   -   -   -   -   -   -   -   -

 4    1    -   - CCC CCC CCC CCC   -   -   -   -   -   -   -   -

 5    1    -   -   -   - CCC CCC   -   -   -   - CCC CCC   -   -

 6    9    -   - CCC CCC CCC CCC   -   -   -   - CCC CCC   -   -

 7    1    -   -   -   - DDD DDD DDD DDD   -   -   -   -   -   -

 8    5    -   -   -   - DDD DDD DDD DDD   -   -   -   -   -   -

 9    1    -   -   -   -   -   - EEE EEE EEE EEE   -   -   -   -

10    5    -   -   -   -   -   - EEE EEE EEE EEE   -   -   -   -

11    4    -   -   -   -   -   -   -   - FFF FFF   -   -   - FFF

12   13    -   -   -   -   -   -   -   - FFF FFF   -   -   - FFF

13    4    -   -   -   -   -   -   -   -   -   - GGG GGG GGG   -

14   11    -   -   -   -   -   -   -   -   -   - GGG GGG GGG   -
```

```
                                            EXAMPLE PROBLEM 2
MODEL RUN ECHO      COMPUTED  0672      DATA GIVEN IN COMPATIBLE UNITS

   SECTION      KFO  JUNCTION ID AND DISTANCE              RK        AK        DK         E          FF       TEMP    PDEL
                               AT ENDS                 REM.RATE  REA.RATE  DEO.RATE  DISP.RATE  ULT/5DAY            MILE-INCR
                                                        (1/DAY)   (1/DAY)   (1/DAY)  (MI2/DAY)            (CENT)    (MI)

 1 GAFF DBLE     1  (BBB,  -5.0)   (DAM, -10.0)        0.376     0.117     0.342      5.000     1.250    28.000   0.500
 2 ECUP DBLE     1  (CCC,   0.0)   (BBB,  -5.0)        0.342     0.117     0.342      5.000     1.250    28.000   1.000
 3 ECJC DBLE     1  (DDD,   5.0)   (CCC,   0.0)        0.342     0.117     0.342     10.000     1.250    28.000   1.000
 4 ECDN DBLE     1  (EEE,   7.5)   (DDD,   5.0)        0.342     0.117     0.342     10.000     1.250    29.000   1.000
 5 USPS DBLE     1  (FFF,  10.0)   (EEE,   7.5)        0.342     0.117     0.342     15.000     1.250    29.000   0.0
 6 PUNC DBLE     1  (CCC,   0.0)   (GGG,  -1.0)        0.370     0.244     0.370      5.000     1.250    30.000   0.250
 7 PBAY S CM     4  (GGG,  -1.0)   (BAY, -10.0)        0.370     0.061     0.370      5.000     1.250    30.000   1.000
 8 ECBY S CM     4  (SEA,  15.0)   (FFF,  10.0)        0.342     0.117     0.342      5.000     1.250    28.000   5.000

   SECTION      CROSS         Q             BD             PR            TIDE1          HT         VOL        WU/AREA*RK
               SEC-AREA      FLOW       BENTHL-DMD     PHOTO-RESP      TIDE-COEF       DEPTH      VOLUME       UNF-LOAD
               (MI**2)   (MI**3/DAY)  (GMS/M2-DAY)   (MG/L-DAY)                      (MTRS)     (MI**3)       (MG/L)

 1 GAFF DBLE  0.359E-04   0.587E-04      0.200E 01      0.0           0.0          0.762E 01   0.0           0.109E-04
 2 ECUP DBLE  0.359E-04   0.587E-04      0.0            0.0           0.0          0.762E 01   0.0           0.0
 3 ECJC DBLE  0.359E-04   0.646E-04      0.0            0.0           0.0          0.762E 01   0.0           0.0
 4 ECDN DBLE  0.359E-04   0.646E-04      0.0            0.0           0.0          0.762E 01   0.0           0.0
 5 USPS DBLE  0.538E-04   0.646E-04      0.0            0.0           0.0          0.914E 01   0.0           0.0
 6 PUNC DBLE  0.359E-05   0.587E-05      0.0            0.0           0.0          0.305E 01   0.0           0.0
 7 PBAY S CM  0.359E-05   0.587E-04      0.0            0.0           0.0          0.122E 02   0.272E-05     0.0
 8 ECBY S CM  0.359E-05   0.646E-04      0.0            0.0           0.500E 00    0.122E 02   0.815E-03     0.0

  SECTION        U          S          V          SD          VD          B           C           G           H        ZZ      FAC

 1 GAFF DBLE  1.63636   0.48309   -0.15581   0.38771   -0.06044   -0.62723   -0.08684   -2.20547   -2.76414    5.18    -1.65
 2 ECUP DBLE  1.63636   0.47219   -0.14492   0.38771   -0.06044    1.43862    0.20615    4.04094    1.49100    0.0     -1.90
 3 ECJC DBLE  1.80000   0.29570   -0.11570   0.23077   -0.05077    0.65495    0.99982    2.98756    3.54438    0.0     -1.90
 4 ECDN DBLE  1.80000   0.29570   -0.11570   0.23077   -0.05077   -0.18599    7.56833   -1.24445   16.75230    0.0     -1.90
```

```
5 USPS DBLE   1.20000   0.19624  -0.11624   0.13701  -0.05701  -0.19956   5.17503  -1.34043  12.52564    0.0    -1.90

6 PUNC DBLE   1.63636   0.48111  -0.15394   0.43848  -0.11120  -7.33666   8.98142 -30.76530  39.19626    0.0    -3.66

7 PBAY 5 CM  16.36363   3.29519  -0.02246   3.27645  -0.00372   5.94027   0.0       2.19518 , 0.0        0.0    -1.50

8 ECBY 5 CM  18.00000   3.61891  -0.01891   3.60649  -0.00650   0.48309   0.0       0.88857   0.0        0.0    -1.90

   EQUATION    JUNCTION ID. SECTIONS                               W              QD          DD          BDD
               MEETING AT JUNCTION.                             PT-LOAD        FLOW-DAM     DEF-DAM     BOD-DAM
               AND DISTANCE                               (M1**3/DAY)(MG/L)  (M1**3/DAY)   (MG/L)      (MG/L)

 1 10 M DM DAM (  1-10.0)(  1-10.0)(  0   0.0)             0.0            0.00006      0.20000     0.20000

 2  1 C 2D BBB (  1 -5.0)(  2 -5.0)(  0   0.0)             0.0            0.0          0.0         0.0

 3  5 M 2D BBB (  1 -5.0)(  2 -5.0)(  0   0.0)             0.0            0.0          0.0         0.0

 4  1 C 2D CCC (  2  0.0)(  3  0.0)(  0   0.0)             0.0            0.0          0.0         0.0

 5  1 C 2D CCC (  6  0.0)(  3  0.0)(  0   0.0)             0.0            0.0          0.0         0.0

 6  9 M3D2 CCC (  2  0.0)(  3  0.0)(  6   0.0)             0.0            0.0          0.0         0.0

 7  1 C 2D DDD (  3  5.0)(  4  5.0)(  0   0.0)             0.00054        0.0          0.0         0.0

 8  5 M 2D DDD (  3  5.0)(  4  5.0)(  0   0.0)             0.00054        0.0          0.0         0.0

 9  1 C 2D EEE (  4  7.5)(  5  7.5)(  0   0.0)             0.0            0.0          0.0         0.0

10  5 M 2D EEE (  4  7.5)(  5  7.5)(  0   0.0)             0.0            0.0          0.0         0.0

11  4 C DC FFF (  5 10.0)(  8 10.0)(  0   0.0)             0.0            0.0          0.0         0.0

12 13 M1CM FFF (  5 10.0)(  8 10.0)(  0   0.0)             0.0            0.0          0.0         0.0

13  4 C DC GGG (  6 -1.0)(  7 -5.0)(  0   0.0)             0.00011        0.0          0.0         0.0

14 11 MCM1 GGG (  6 -1.0)(  7 -5.0)(  0   0.0)             0.00011        0.0          0.0         0.0
```

```
                                     EXAMPLE PROBLEM 2
        ECHO                      COMPUTED      0672

  SECT.          SECT.            DIST.           BOD           D.O DEFICIT        JUNCTION
   NO.           NAME              MI             MG/L             MG/L               ID
  -----          -----            -----          ------         -----------        --------

    1            GAFF            -10.0           0.3890           0.7693             DAM

    1            GAFF             -9.5           0.4185           0.8603

    1            GAFF             -9.0           0.4454           0.9503

    1            GAFF             -8.5           0.4896           1.0376

    1            GAFF             -8.0           0.4913           1.1220

    1            GAFF             -7.5           0.5103           1.2028

    1            GAFF             -7.0           0.5267           1.2795

    1            GAFF             -6.5           0.5402           1.3515

    1            GAFF             -6.0           0.5507           1.4181

    1            GAFF             -5.5           0.5579           1.4785

    1            GAFF             -5.0           0.5612           1.5318             BBB
```

```
                                EXAMPLE PROBLEM 2
      ECHO                   COMPUTED     0672

SECT.         SECT.         DIST.          BOD          D.O DEFICIT        JUNCTION
 NO.          NAME           MI           MG/L             MG/L               ID
-----         -----         -----        ------        -----------        --------

  2           ECUP          -5.0         0.5612          1.5318              BBB

  2           ECUP          -4.0         0.5857          1.6424

  2           ECUP          -3.0         0.6673          1.7816

  2           ECUP          -2.0         0.8350          1.9562

  2           ECUP          -1.0         1.1355          2.1676

  2           ECUP           0.0         1.6448          2.4052              CCC
```

```
                                        EXAMPLE PROBLEM 2
        ECHO                         COMPUTED      0672

   SECT.          SECT.          DIST.          BOD          D.O DEFICIT        JUNCTION
    NO.           NAME            MI            MG/L            MG/L               ID
   -----          -----          -----         ------        -----------       --------

     3            ECJC            0.0          1.6448          2.4052             CCC
     3            ECJC            1.0          1.7620          2.5229
     3            ECJC            2.0          1.9665          2.6133
     3            ECJC            3.0          2.2898          2.6625
     3            ECJC            4.0          2.7606          2.6479
     3            ECJC            5.0          3.4279          2.5347             DDD
```

```
                                  EXAMPLE PROBLEM 2
      ECHO                      COMPUTED     0672

SECT.          SECT.          DIST.          BOD          D.O DEFICIT       JUNCTION
 NO.           NAME            MI            MG/L            MG/L              ID
-----          -----          -----         ------       -----------       ---------

  4            ECDN            5.0          3.4279         2.5347             DDD
  4            ECDN            6.0          2.6835         2.2823
  4            ECDN            7.0          1.8932         1.8832
  4            ECDN            7.5          1.4691         1.6295             EEE
```

```
                                  EXAMPLE PROBLEM 2
     ECHO                     COMPUTED     0672

SECT.         SECT.         DIST.          BOD        D.O DEFICIT      JUNCTION
 NO.          NAME           MI           MG/L           MG/L             ID
-----         -----         -----        ------      -----------      --------

  5           USPS           7.5         1.4691         1.6295           EEE

  5           USPS           8.5         1.0809         1.3649

  5           USPS           9.5         0.6860         1.0572

  5           USPS          10.0         0.4931         0.8889           FFF
```

```
                                   EXAMPLE PROBLEM 2
      ECHO                      COMPUTED      C672

   SECT.           SECT.           DIST.            BOD          D.O DEFICIT         JUNCTION
    NO.            NAME             MI             MG/L             MG/L                ID
   -----           -----           -----          ------        ------------        --------

     6             PUNC            -1.0           5.9403           2.1997              GGG

     6             PUNC            -0.8           4.9655           2.2709

     6             PUNC            -0.5           3.9315           2.3254

     6             PUNC            -0.3           2.8283           2.3683

     6             PUNC             0.0           1.6440           2.4052              CCC
     ECHO                        EXAMPLE PROBLEM 2
                              COMPUTED     0672

  SECT.          SECT.          DIST.          BOD          D.O DEFICIT          JUNCTION
   NO.           NAME            MI            MG/L             MG/L                ID
  -----          -----          -----         ------        -----------          --------

    7            PBAY           -10.0         5.9403          2.1998               BAY
    7            PBAY            -9.0         5.9403          2.1998
    7            PBAY            -8.0         5.9403          2.1998
    7            PBAY            -7.0         5.9403          2.1998
    7            PBAY            -6.0         5.9403          2.1998
    7            PBAY            -5.0         5.9403          2.1998
    7            PBAY            -4.0         5.9403          2.1998
    7            PBAY            -3.0         5.9403          2.1998
    7            PBAY            -2.0         5.9403          2.1998
    7            PBAY            -1.0         5.9403          2.1998               GGG
```

```
                                          EXAMPLE PROBLEM 2
        ECHO                          COMPUTED      0672

  SECT.           SECT.           DIST.          BOD          D.O DEFICIT        JUNCTION
   NO.            NAME             MI            MG/L            MG/L               ID
  -----           -----           -----         ------        -----------        --------

    8             ECRY            10.0          0.4831          0.8889              FFF
    8             ECBY            15.0          0.4831          0.8889              SEA
```

References

1. Hydroscience, Inc.: Mathematical Models for Water Quality for the Hudson-Champlain and Metropolitan Coastal Water Pollution Control Project, FWPCA. Hydroscience, Inc., Westwood, N.J., April 1968.

2. Thomann, R.V., Systems Analysis and Water Quality Management, Environmental Science Service Division, N.Y., N.Y., 1971, p.p. 123-190.

3. O'Connor, D.J., Oxygen Balance of an Estuary. Jour. San. Eng. Div. ASCE, Vol 86, No. SA3, May 1960, p.p. 35-55.

4. O'Connor, D.J., St. John, J.P., and Di Toro, D.M., Water Quality Analysis of the Delaware River Estuary. Jour. San. Eng. Div. ASCE, Vol 94, No. SA6, Dec. 1968, p.p. 1225-1252.

5. Tracor, Inc.: Estuarine Modelling: An Assessment, Prepared for Water Quality Office, Environmental Protection Agency. Tracor, Inc., Austin, Texas, February 1971, p.p. 102-168.

6. O'Connor, D.J. and Di Toro, D.M., "The Solution of the Continuity Equation in Cylindrical Coordinates with Dispersion and Advection for an Instantaneous Release." Proceedings-Symposium on Diffusion in Oceans and Fresh Waters, Lamont Geological Observatory. Palisades, New York, 1965.

7. O'Connor, D.J., The Temporal and Spatial Distribution of Dissolved Oxygen in Streams. Water Resources Research, Vol 3, No. 1, 1967, p.p. 65-79.

8. "Effects of Pollution Discharges on the Thames Estuary", Water Pollution Research, Technical Paper No. 11, Dept. of Sci. and Ind. Res., Her Majesty's Stationary Office, 1964, 609 p.p. + xxvii.

9. Courchaine, R.J., Significance of Nitrification in Stream Analysis - Effects on the Oxygen Balance, Journal Water Poll. Control Fed., Vol 40, No. 5, Part 1, May 1968, p.p. 835-847.

10. Daily J. E. and Harleman D.R.F. Numerical Model for the Prediction of Transient Water Quality in Estuary Networks published by Md.T October 1972.

11. O'Connor, D. J. and Mancini, J. "Water Quality Analysis of the New York Harbor Complex" Journal WPCF, November 1972.

Appendix

ES001 Source Deck

(IBM 370)

```
FORTRAN IV G LEVEL  20                MAIN              DATE = 72258         14/03/41                   PAGE 0001

                                                                                            MAIN 000
 0001            REAL*8 IDJOB,IDATE,ID,IDJ                                                  MAIN 001
 0002            COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100),          MAIN 002
                1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),              MAIN 003
                2S(100),JMAX,IS1,ISC(100),CK(100),NJU(100),XJU(100),KFO(100),               MAIN 004
                3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100),           MAIN 005
                4JND(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),MAIN 006
                5PEEL(100),JDV(100),QD(100),DD(100),BDD(100),                               MAIN 007
                6ID(100),ICJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM            MAIN 008
 0003            COMMON/NEW/FAC1,FAC2,FAC3,ITYPE(35)                                        MAIN 009
 0004            DIMENSION BM(100,100),DM(100,100),BMI(100,100),DMI(100,100)                MAIN 010
 0005            DIMENSION BFOR(100),BSOL(100),DFOR(100),DSOL(100),BBX1(250)                MAIN 011
 0006            DIMENSION DDX1(250),MI(100),MJ(100)                                        MAIN 012
 0007            EXTERNAL FR,FBR,FPR,FBPR,YD,YB,FA,FBA,FPA,FBPA                             MAIN 013
 0008            JBM=100                                                                    MAIN 014
 0009          1 READ(5,10,END=999)IDJOB,IDATE,IS2,IS1,NN,IND,ICON,FAC1,FAC2,FAC3,          MAIN 015
                1ITYPE                                                                      MAIN 016
 0010            DO 26 I=1,250                                                              MAIN 017
 0011            BBX1(I)=0.                                                                 MAIN 018
 0012         26 DDX1(I)=0.                                                                 MAIN 019
 0013            GO TO 25                                                                   MAIN 020
 0014         24 READ(5,10,END=999)IDJOB,IDATE,IS2,IS1,NN,IND,NOTM,FAC1,FAC2,FAC3,          MAIN 021
                1ITYPE                                                                      MAIN 022
 0015         10 FORMAT(A4,1X,A4,1X,5I3,F5.0,2F6.0,3X,35A1)                                 MAIN 023
 0016         25 PRINT 157,IDJOB,IDATE,IS2,IS1,NN,IND,ICON,FAC1,FAC2,FAC3,ITYPE             MAIN 024
 0017        157 FORMAT('1JOB ID AND DATE CARD'/1X,20(1H-)/1X,A8,1X,A8,5I5,3F8.3,           MAIN 025
                11X,35A1)                                                                   MAIN 026
 0018            IMAX=IS2+IS1                                                               MAIN 027
 0019            JMAX=2*IS2+IS1                                                             MAIN 028
 0020            CALL DATAR(NFLAG,MI,MJ)                                                    MAIN 029
 0021            PRINT 40,ITYPE,IDJOB,IDATE                                                 MAIN 030
 0022         30 FORMAT('1',42X,35A1/' MODEL RUN ',A8,2X,'COMPUTED',2X,A8,2X                MAIN 031
                1'DATA GIVEN IN COMPATIBLE UNITS'/)                                         MAIN 032
 0023         40 FORMAT('1',42X,35A1/' MODEL RUN ',A8,'  COMPUTED',2X,A8,2X,                MAIN 033
                130HDATA GIVEN IN  ORIGINAL  UNITS/)                                        MAIN 034
 0024            CALL PRI(MI,MJ,0)                                                          MAIN 035
 0025            CALL PRJ(MI,MJ,0)                                                          MAIN 036
 0026            CALL DATAC(ICON)                                                           MAIN 037
 0027            IF(NFLAG) 6,6,5                                                            MAIN 038
 0028          5 PRINT 20,NFLAG                                                             MAIN 039
 0029         20 FORMAT(1H0,I5,2X,31HERRORS FOUND COULD NOT CONTINUE /)                     MAIN 040
 0030            GO TO 1                                                                    MAIN 041
 0031          6 IF(NN) 4,4,14                                                              MAIN 042
 0032          4 IJK=1                                                                      MAIN 043
 0033            CALL MATS(FR,FBR,FPR,FBPR,IJK,BM)                                          MAIN 044
 0034            CALL BRIGHT(BFOR)                                                          MAIN 045
 0035            DO 2 I=1,JBM                                                               MAIN 046
 0036            BSOL(I)=BFOR(I)                                                            MAIN 047
 0037            DO 2 J=1,JBM                                                               MAIN 048
 0038            BMI(I,J)=BM(J,I)                                                           MAIN 049
 0039          2 CONTINUE                                                                   MAIN 050
 0040            CALL MATINV(BMI,JMAX,BSOL,1,BET,JMAX,IND,1)                                MAIN 051
 0041            CALL BCGH(BSOL,IS2,JMAX,KFO,B,C)
```

```
FORTRAN IV G LEVEL  20                MAIN               DATE = 72258          14/03/41              PAGE 0002

      0042                IF(ICON-1)21,22,21                                                 MAIN 052
      0043             21 IJK=2                                                              MAIN 053
      0044                CALL MATS(FA,FBA,FPA,FBPA,IJK,DM)                                  MAIN 054
      0045                CALL DRIGHT    (DFOR)                                              MAIN 055
      0046                DO3I=1,JBM                                                         MAIN 056
      0047                DSOL(I)=DFOR(I)                                                    MAIN 057
      0048                DO 3 J=1,JBM                                                       MAIN 058
      0049                DMI(I,J) = DM(J,I)                                                 MAIN 059
      0050              3 CONTINUE                                                           MAIN 060
      0051                CALL MATINV(DMI,JMAX,DSOL,1,DET,JMAX,IND,2)                        MAIN 061
      0052             11 CALL BCGH(DSOL,IS2,JMAX,KFO,G,H)                                   MAIN 062
      0053             22 PRINT 30,ITYPE,IDJOB,IDATE                                         MAIN 063
      0054                CALL PRI(MI,MJ,1)                                                  MAIN 064
      0055                CALL PRII(MI,MJ,1)                                                 MAIN 065
      0056                CALL PRJ(MI,MJ,1)                                                  MAIN 066
      0057             27 N=0                                                                MAIN 067
      0058                DO 13 I=1,IMAX                                                     MAIN 068
      0059             13 CALL HDOUT(I,N,ICON,BBX1,DDX1)                                     MAIN 069
      0060                IF(ICON-2)1,23,23                                                  MAIN 070
      0061             23 ICON=ICON+1                                                        MAIN 071
      0062                GO TO(1,1,24,27,1),ICON                                            MAIN 072
      0063             14 IF(ICON-1)28,28,1                                                  MAIN 073
      0064             28 CALL BRIGHT(BFOR)                                                  MAIN 074
      0065                CALL MAMS(BMI,BFOR,BSOL,JMAX,JMAX,1,100)                           MAIN 075
      0066                CALL BCGH(BSOL,IS2,JMAX,KFO,B,C)                                   MAIN 076
      0067                IF(ICON-1)29,22,1                                                  MAIN 077
      0068             29 CALL DRIGHT(DFOR)                                                  MAIN 078
      0069                CALLMAMS(DMI,DFOR,DSOL,JMAX,JMAX,1,100)                            MAIN 079
      0070                GO TO 11                                                           MAIN 080
      0071            999 STOP                                                               MAIN 081
      0072                END                                                                MAIN 082
```

```
FORTRAN IV G LEVEL  20                 BCGH              DATE = 72258         14/03/41                 PAGE 0001

      0001              SUBROUTINE BCGH(SOL,IS2,JMAX,KFO,BG,CE)                              BCGH 000
      0002              DIMENSION SOL(1),BG(1),CE(1),KFO(1)                                  BCGH 001
      0003              IS1 = JMAX - 2 * IS2                                                 BCGH 002
      0004              DO1N=1,IS2                                                           BCGH 003
      0005              N2=N*2                                                               BCGH 004
      0006              CE(N)=SOL(N2)                                                        BCGH 005
      0007              M=N2-1                                                               BCGH 006
      0008              BG(N)=SOL(M)                                                         BCGH 007
      0009            1 CONTINUE                                                             BCGH 008
      0010              DO2I=1,IS1                                                           BCGH 009
      0011              K=I+2*IS2                                                            BCGH 010
      0012              N=I+IS2                                                              BCGH 011
      0013              KF=KFO(N)                                                            BCGH 012
      0014              GO TO(2,12,13,12),KF                                                 BCGH 013
      0015           12 BG(N)=SOL(K)                                                         BCGH 014
      0016              CE(N)=0.0                                                            BCGH 015
      0017              GO TO 2                                                              BCGH 016
      0018           13 CE(N)=SOL(K)                                                         BCGH 017
      0019              BG(N)=0.0                                                            BCGH 018
      0020            2 CONTINUE                                                             BCGH 019
      0021              RETURN                                                               BCGH 020
      0022              END                                                                  BCGH 021
```

```
FORTRAN IV G LEVEL  20              BDOUT              DATE = 72258          14/03/41                PAGE 0001

   0001              SUBROUTINE BDOUT(I,N,ICON,BBX1,DDX1)                                    BDOUT000
   0002              REAL*8 IDJOB,IDATE,ID,IDJ                                               BDOUT001
   0003              COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100),       BDOUT002
                    1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),           BDOUT003
                    2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),KFO(100),            BDOUT004
                    3V(100),SD(100),B(100),Z(100),AREA(100),TEMP(100),E(100),HT(100),        BDOUT005
                    4JNO(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),      BDOUT006
                    5PDEL(100),JBV(100),GD(100),DD(100),BDD(100),                            BDOUT007
                    6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM         BDOUT008
   0004              COMMON/NEW/FAC1,FAC2,FAC3,ITYPE(35)                                     BDOUT009
   0005              DIMENSION BBX1(1),DDX1(1)                                               BDOUT010
   0006              IF(ICON-4)40,41,41                                                      BDOUT011
   0007           40 PRINT 30,ITYPE,IDJOB,IDATE                                              BDOUT012
   0008              GO TO 44                                                                BDOUT013
   0009           41 PRINT 42                                                                BDOUT014
   0010           42 FORMAT('1',40X,'COUPLED SYSTEM OF PREVIOUS 2 COMPONENTS'                BDOUT015
   0011              PRINT 43,IDJOB,IDATE                                                    BDOUT016
   0012           43 FORMAT(10X,A8,21X8HCOMPUTED 5XA8//1H0,2X,'SECT.',14X,                   BDOUT017
                    1'SECT.',15X,5HDIST.,15X 3HBOD,13X4HD.O ,'DEFICIT',12X,'JUNCTION'/       BDOUT018
                    24X,'NO.',15X,'NAME',16X,' MI ',14X,' MG/L ',14X,' MG/L ',17X,'ID'/      BDOUT019
                    33X,'-----',14X,'-----',14X,'-----',13X,'------',13X,11(1H-),12X,        BDOUT020
                    48(1H-))                                                                 BDOUT021
   0013           44 DELTA=1.0                                                               BDOUT022
   0014              IF(PDEL(I)) 5,5,4                                                       BDOUT023
   0015            4 DELTA = PDEL(I)                                                         BDOUT024
   0016            5 XS=XJD(I)                                                               BDOUT025
   0017              XE=XJU(I)                                                               BDOUT026
   0018              JJS=NJU(I)                                                              BDOUT027
   0019              JJE=NJD(I)                                                              BDOUT028
   0020              CALL BODO(I,XE,BBX,DDX)                                                 BDOUT029
   0021              N=N+1                                                                   BDOUT030
   0022              IF(ICON-4)6,7,7                                                         BDOUT031
   0023            6 DDX1(N)=DDX1(N)+DDX                                                     BDOUT032
   0024              BBX1(N)=BBX1(N)+BBX                                                     BDOUT033
   0025              PRINT 20, I,ID(I),XE,BBX,DDX,JJS                                        BDOUT034
   0026              GO TO 8                                                                 BDOUT035
   0027            7 PRINT 20, I,ID(I),XE,BBX1(N),DDX1(N),JJS                                BDOUT036
   0028           20 FORMAT(1H0,I5,16XA8,F16.1,2F20.4,17XA3)                                 BDOUT037
   0029            8 XE=XE+DELTA                                                             BDOUT038
   0030              IF(XE-XS)9,10,10                                                        BDOUT039
   0031            9 CALL BODO(I,XE,BBX,DDX)                                                 BDOUT040
   0032              N=N+1                                                                   BDOUT041
   0033              IF(ICON-4)32,14,14                                                      BDOUT042
   0034           32 BBX1(N)=BBX1(N)+BBX                                                     BDOUT043
   0035              DDX1(N)=DDX1(N)+DDX                                                     BDOUT044
   0036           31 PRINT 20, I,ID(I),XE,BBX,DDX                                            BDOUT045
   0037              GO TO 8                                                                 BDOUT046
   0038           14 PRINT 20, I,ID(I),XE,BBX1(N),DDX1(N)                                    BDOUT047
   0039              GO TO 8                                                                 BDOUT048
   0040           10 CALL BODO(I,XS,BBX,DDX)                                                 BDOUT049
   0041              N=N+1                                                                   BDOUT050
   0042              IF(ICON-4)16,17,17                                                      BDOUT051
```

```
    0043              16 DDX1(N)=DDX1(N)+DDX                                                BDOUT052
    0044                 BBX1(N)=BBX1(N)+BBX                                                BDOUT053
    0045                 PRINT 20,I,ID(I),XS,BBX,DDX,JJE                                    BDOUT054
    0046                 GO TO 18                                                           BDOUT055
    0047              17 PRINT 20, I,ID(I),XS,BBX1(N),DDX1(N),JJE                           BDOUT056
    0048              30 FORMAT(1H1,43X,35A1/10X,A8,21X8HCOMPUTED 5XA8//1H0,2X,'SECT.',14X,BDOUT057
                        1'SECT.',15X,5HDIST.,15X 3HBOD,13X4HD.O ,'DEFICIT',12X,'JUNCTION'/ BDOUT058
                        24X,'NO.',15X,'NAME',16X,' MI ',14X,' MG/L ',14X,' MG/L ',17X,'ID'/BDOUT059
                        33X,'-----',14X,'-----',14X,'-----',13X,'------',13X,11(1H-),12X,  BDOUT060
                        48(1H-))                                                           BDOUT061
    0049              18 RETURN                                                             BDOUT062
    0050                 END                                                                BDOUT063
```

```
 0001            SUBROUTINE BODO(I,X,BBB,DDD)                                       BODO 000
 0002            REAL*8 IDJOB,IDATE,ID,IDJ                                          BODO 001
 0003            COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100),  BODO 002
                1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),      BODO 003
                2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),KFO(100),       BODO 004
                3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100),   BODO 005
                4JND(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),BODO 006
                5PDEL(100),JBV(100),QD(100),DD(100),BDD(100),                       BODO 007
                6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM    BODO 008
 0004            EXTERNAL  FR,FBR,FA,FBA                                            BODO 009
 0005            BBM=DM(KFO,B,C,FR,FBR,I,X)                                         BODO 010
 0006            YDIX=ZZ(I)+FAC(I)*BBM                                              BODO 011
 0007            BBB=BBM+YB(I,X)                                                    BODO 012
 0008            DDM=DM(KFO,G,H,FA,FBA,I,X)                                         BODO 013
 0009            IF(KFO(I)-4)1,2,2                                                  BODO 014
 0010          1 DDD=DDM+YDIX                                                       BODO 015
 0011            GO TO 3                                                            BODO 016
 0012          2 DDD=DDM                                                            BODO 017
 0013          3 CONTINUE                                                           BODO 018
 0014            RETURN                                                             BODO 019
 0015            END                                                                BODO 020
```

```
FORTRAN IV G LEVEL  20              BRIGHT              DATE = 72258         14/03/41            PAGE 0001

 0001            SUBROUTINE BRIGHT (FOR)                                                  BRIGH000
 0002            DIMENSION FOR(1)                                                         BRIGH001
 0003            REAL*8 IDJOB,IDATE,ID,IDJ                                                BRIGH002
 0004            COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100),        BRIGH003
                1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),            BRIGH004
                2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),KFD(100),             BRIGH005
                3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100),         BRIGH006
                4JNO(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),       BRIGH007
                5PDEL(100),JBV(100),QD(100),DD(100),BDD(100),                             BRIGH008
                6IE(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM          BRIGH009
 0005            DO 101   I=1,JBM                                                         BRIGH010
 0006        101 FOR(I)=0.0                                                               BRIGH011
 0007            DO991J=1,JMAX                                                            BRIGH012
 0008            JB=JBV(J)                                                                BRIGH013
 0009            I =ISA(J)                                                                BRIGH014
 0010            X=XSA(J)                                                                 BRIGH015
 0011            K =ISB(J)                                                                BRIGH016
 0012            Y=XSB(J)                                                                 BRIGH017
 0013            KK =ISC(J)                                                               BRIGH018
 0014            Z=XSC(J)                                                                 BRIGH019
 0015            GO TO (1,1,1,4,5,5,5,8,9,10,11,11,13,13),JB                              BRIGH020
 0016          1 FOR(J)=YB(K,Y)- YB(I,X)                                                  BRIGH021
 0017            GO TO 99                                                                 BRIGH022
 0018          4 FOR(J) = -YB(I,X)                                                        BRIGH023
 0019            GO TO 99                                                                 BRIGH024
 0020          5 FOR(J) = W(J)+Q(I)*YB(I,X)-Q(K)*YB(K, Y )                                BRIGH025
 0021            GO TO 99                                                                 BRIGH026
 0022          8 FOR(J)=W(J)+Q(I)*YB(I,X)-Q(K)*YB(K,Y)-Q(KK)*YB(KK,Z)                     BRIGH027
 0023            GO TO 99                                                                 BRIGH028
 0024          9 FOR(J)=W(J)+Q(I)*YB(I,X)+Q(KK)*YB(KK,Z)-Q(K)*YB(K,Y)                     BRIGH029
 0025            GO TO 99                                                                 BRIGH030
 0026         10 FOR(J)=QD(J)*BDD(J)  -Q(K)*YB(K,Y)                                       BRIGH031
 0027            GO TO 99                                                                 BRIGH032
 0028         11 FOR(J) =W (J)-Q(I)*YB(I,X)                                               BRIGH033
 0029            IF(JB-11)99,99,12                                                        BRIGH034
 0030         12 FOR(J)=FOR(J)  -Q(K )*YB(K ,Y)                                           BRIGH035
 0031            GO TO 99                                                                 BRIGH036
 0032         13 FOR(J)=W(J )+Q(I)*YB(I,X)                                                BRIGH037
 0033            IF(JB-13)99,99,14                                                        BRIGH038
 0034         14 FOR(J) =    FOR(J)            + Q(K )*YB(K ,Y)                           BRIGH039
 0035         99 CONTINUE                                                                 BRIGH040
 0036        991 CONTINUE                                                                 BRIGH041
 0037            RETURN                                                                   BRIGH042
 0038            END                                                                      BRIGH043
```

```
0001                SUBROUTINE DATAC(ICON)                                                DATAC000
0002                REAL*8 IDJOB,IDATE,ID,IDJ                                             DATAC001
0003                COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100),   DATAC002
                   1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),        DATAC003
                   2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),KFO(100),         DATAC004
                   3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100),    DATAC005
                   4JNO(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),DATAC006
                   5PUEL(100),JBV(100),CD(100),DD(100),BDD(100),                        DATAC007
                   6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM     DATAC008
0004                COMMON/NEW/FAC1,FAC2,FAC3,ITYPE(35)                                  DATAC009
0005                DIMENSION LINE(100)                                                   DATAC010
0006                DATA LITERL/3H  -/                                                    DATAC011
0007                DO55I=1,IMAX                                                          DATAC012
0008                AK(I)=AK(I)*FAC1**(TEMP(I)-20.)                                       DATAC013
0009                DK(I)=DK(I)*FAC2**(TEMP(I)-20.)                                       DATAC014
0010                RK(I)=RK(I)*FAC3**(TEMP(I)-20.)                                       DATAC015
0011             55 CONTINUE                                                              DATAC016
0012                CA=1.0/(5280.0**2)                                                    DATAC017
0013                CV=1./(5280.**3)                                                      DATAC018
0014                CF=3600.0*24.0/(5280.**3)                                             DATAC019
0015                CL=(1.0E+6)/(62.4*5280.**3)                                           DATAC020
0016                CM=12.0/39.37                                                         DATAC021
0017                DO5I=1,JMAX                                                           DATAC022
0018                IF(I-IMAX)60,60,70                                                    DATAC023
0019             60 AREA(I)=AREA(I)*CA                                                    DATAC024
0020                Q(I)=Q(I)*CF                                                          DATAC025
0021                WU(I)=WU(I)*CL                                                        DATAC026
0022                HT(I)=HT(I)*CM                                                        DATAC027
0023                VOL(I)=VOL(I)*CV                                                      DATAC028
0024             70 W(I)=W(I)*CL                                                          DATAC029
0025                QD(I)=QD(I)*CF                                                        DATAC030
0026              5 CONTINUE                                                              DATAC031
0027                DO32 I=1,IMAX                                                         DATAC032
0028                FAC(I)=DK(I)*FF(I)/(AK(I)-RK(I))                                      DATAC033
0029                ZZ(I)=0.0                                                             DATAC034
0030             31 AA=BD(I)/HT(I)                                                        DATAC035
0031                IF(RK(I))14,14,12                                                     DATAC036
0032             12 BB=DK(I)*WU(I)*FF(I)/(RK(I)*AREA (I))                                 DATAC037
0033                GO TO 15                                                              DATAC038
0034             14 BB=0.                                                                 DATAC039
0035             15 ZZ(I)=(AA*PR(I)+BB)/AK(I)                                             DATAC040
0036             32 CONTINUE                                                              DATAC041
0037                DO 1 I=1,IMAX                                                         DATAC042
0038                U(I)=Q(I)/AREA(I)                                                     DATAC043
0039                CALL SVD(U(I),E(I),AK(I),SD(I),VD(I))                                 DATAC044
0040                CALL SVD(U(I),S(I),RK(I),S(I),V(I))                                   DATAC045
0041              1 CONTINUE                                                              DATAC046
0042                PRINT 20,     (J,J=1,JMAX)                                            DATAC047
0043             20 FORMAT('1',38X,'SYMBOLIC DO DEFICIT AND BOD MATRIX'/4X,'ROW',2X,     DATAC048
                   1'JBV',24I4/13X,24I4)                                                 DATAC049
0044            501 IS2M=IS2*2                                                            DATAC050
0045                DO 3 JJ=1,JMAX                                                        DATAC051
```

```
0046              J=JND(JJ)                                                    DATAC052
0047              DO 2 L=1,JBM                                                 DATAC053
0048            2 LINE(L)=LITERL                                               DATAC054
0049              CALL PLACE(J,JJ,IS2M,ISA,LINE)                               DATAC055
0050              CALL PLACE(J,JJ,IS2M,ISB,LINE)                               DATAC056
0051              CALL PLACE(J,JJ,IS2M,ISC,LINE)                               DATAC057
0052              PRINT 30,JJ,JBV(JJ),(LINE(I),I=1,JMAX)                       DATAC058
0053           30 FORMAT(1HO,2I5,2X,24(1XA3)/13X,24(1X,A3))                    DATAC059
0054            3 CONTINUE                                                     DATAC060
0055              IF(ICON-1)11,8,11                                            DATAC061
0056            8 DO 9 I=1,JMAX                                                DATAC062
0057              DD(I)=0.                                                     DATAC063
0058              SD(I)=0.                                                     DATAC064
0059              VD(I)=0.                                                     DATAC065
0060              G(I)=0.                                                      DATAC066
0061              H(I)=0.                                                      DATAC067
0062              FAC(I)=0.                                                    DATAC068
0063            9 ZZ(I)=0.                                                     DATAC069
0064           11 RETURN                                                       DATAC070
0065              END                                                          DATAC071
```

```
FORTRAN IV G LEVEL  20              DATAR              DATE = 72258          14/03/41                PAGE 0001

  0001                SUBROUTINE DATAR(NFLAG,MI,MJ)                                               DATAR000
  0002                REAL*8 IDJOB,IDATE,ID,IDJ                                                   DATAR001
  0003                COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100),           DATAR002
                     1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),               DATAR003
                     2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),KFO(100),                DATAR004
                     3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100),            DATAR005
                     4JNO(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),DATAR006
                     5PDEL(100),JBV(100),QD(100),DD(100),BDD(100),                                DATAR007
                     6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM             DATAR008
  0004                COMMON/NEW/FAC1,FAC2,FAC3,ITYPE(35)                                         DATAR009
  0005                DIMENSION IDJB(3,14),ISV(3),IDHJB(14),IDKFO(6),MI(1),MJ(1)                  DATAR010
  0006                DIMENSION LI(3),NDA(3)                                                      DATAR011
  0007                DATA IDHJB/'C 2D','C DN','C UP','C DC','M 2D','M DN','M DP',                DATAR012
                     1'M3C1','M3D2','M DM','MCM1','MCM2','M1CM','M2CM'/                           DATAR013
  0008                DATA IDKFO/'DBLE','S NI','S PI','S CM','MISS','N CL'/                       DATAR014
  0009                DATA IPASS/'PASS'/,IFAIL/'FAIL'/,IBLNK/'    '/                              DATAR015
  0010                DATA IDJB/2*1,5,2,1,5,1,3,5,1,4,5,2*1 ,5,2,1,5,1,3,5,                       DATAR016
                     26*1,1,1,5,1,4,5,2*1,4,1,4,5,2*1,4/                                          DATAR017
  0011                NFLAG=0                                                                     DATAR018
  0012                KNFLB=IBLNK                                                                 DATAR019
  0013                PRINT 158                                                                   DATAR020
  0014            158 FORMAT(' INPUT SECTION CARD LISTING'/1X,26(1H-))                            DATAR021
  0015                DO 6 II=1,IMAX                                                              DATAR022
  0016                READ10,NS2, I,NUSD,NJU(I),XJU(I),AREA(I),RK(I),AK(I),FF(I),Q(I),            DATAR023
                     1BD(I),WU(I),TIDE1(I),IDJ                                                    DATAR024
  0017                PRINT20,NS2,I,NUSD,NJU(I),XJU(I),AREA(I),RK(I),AK(I),FF(I),Q(I),            DATAR025
                     1BD(I),WU(I),TIDE1(I),IDJ                                                    DATAR026
  0018              2 READ 10,NS1,I,KFO(I),NJD(I),XJD(I),UNUSED,DK(I),E(I),TEMP(I),PR(I)DATAR027
                     1,HT(I),VOL(I),PDEL(I),ID(I)                                                 DATAR028
  0019                PRINT20,NS1,I,KFO(I),NJD(I),XJD(I),UNUSED,DK(I),E(I),TEMP(I),PR(I)DATAR029
                     1,HT(I),VOL(I),PDEL(I),ID(I)                                                 DATAR030
  0020             40 FORMAT( 28HOSECTION CARDS NOT IN ORDER.,2I10//)                             DATAR031
  0021                IST=NTPS(I)                                                                 DATAR032
  0022                IF(IST.GT.4) GO TO 71                                                       DATAR033
  0023                GO TO 14                                                                    DATAR034
  0024             71 NFLAG=NFLAG+1                                                               DATAR035
  0025                PRINT 11,KFO(I),I                                                           DATAR036
  0026             11 FORMAT(' ERROR IN KFO(',I2,') FOR SECTION',I3)                              DATAR037
  0027             14 IF(AK(I).EQ.RK(I)) GO TO 73                                                 DATAR038
  0028                GO TO 74                                                                    DATAR039
  0029            700 FORMAT(8HOSECTION,I5,2X,2HIS,2X,A8/)                                        DATAR040
  0030             73 NFLAG=NFLAG+1                                                               DATAR041
  0031                PRINT 701,I,AK(I),RK(I)                                                     DATAR042
  0032            701 FORMAT(24HOERROR IN AK I .OR.RK I ,2X I10,2E20.4)                           DATAR043
  0033             74 IF(HT(I)) 6,75,6                                                            DATAR044
  0034             75 NFLAG=NFLAG+1                                                               DATAR045
  0035                PRINT 702,I,HT(I)                                                           DATAR046
  0036            702 FORMAT(16HOERROR IN HT I  ,9X,I10,2E20.4)                                   DATAR047
  0037              6 CONTINUE                                                                    DATAR048
  0038                DO 72 II=1,IMAX                                                             DATAR049
  0039                IST=NTPS(II)                                                                DATAR050
  0040                MI(II)=IDKFO(IST)                                                           DATAR051
```

```
   72 PRINT 700,II,IDKFO(IST)                                           DATAR052
      PRINT 159                                                         DATAR053
  159 FORMAT(' INPUT JUNCTION CARD LISTING'/1X,27(1H-))                 DATAR054
      DO 8 JJ=1,JMAX                                                    DATAR055
   10 FORMAT(2X,2I3,I2, A3  ,F5.0,F7.0,3F5.0,4F8.0,4X,A4)               DATAR056
   20 FORMAT(1H0,3I4,3X,A3,F10.3,1X,F9.1,3F10.3,F9.1,F10.3,1X,E10.3,    DATAR057
     1F10.3,5X,A4/)                                                     DATAR058
      READ 50,J,JNO(J),JBV(J),ISA(J),XSA(J),ISB(J),XSB(J),ISC(J),XSC(J),DATAR059
     1W(J),QD(J),DD(J),BDD(J),IDJ                                       DATAR060
      PRINT 60,J,JNO(J),JBV(J),ISA(J),XSA(J),ISB(J),XSB(J),ISC(J),XSC(J)DATAR061
     1,W(J),QD(J),DD(J),BDD(J),IDJ                                      DATAR062
   50 FORMAT(1X,I2,A3,I3,3(I3,F6.1),4F6.0,16X,A4)                       DATAR063
   60 FORMAT(1H0,I4,1X,A3,I5, 3(1X,1H ,I5,F8.2,1H ,1X),4F8.1,18X,A4/)   DATAR064
    8 CONTINUE                                                          DATAR065
      DO 4 J=1,JMAX                                                     DATAR066
      JB=JBV(J)                                                         DATAR067
      MJ(J)=IDHJB(JB)                                                   DATAR068
      IZ=0                                                              DATAR069
      DO 86 LL=1,3                                                      DATAR070
      GO TO (81,82,83),LL                                               DATAR071
   81 IS=ISA(J)                                                         DATAR072
      GO TO 84                                                          DATAR073
   82 IS=ISB(J)                                                         DATAR074
      GO TO 84                                                          DATAR075
   83 IS=ISC(J)                                                         DATAR076
   84 ISV(LL)=IS                                                        DATAR077
      NTY=NTPS(IS)                                                      DATAR078
      IDT=IPASS                                                         DATAR079
      LI(LL)=IDKFO(NTY)                                                 DATAR080
      NDA(LL)=KNALB                                                     DATAR081
      ILB=IDJB(LL,JB)                                                   DATAR082
      IF(NTY-ILB) 85,1,85                                               DATAR083
    1 IF(LL-3)86,5,5                                                    DATAR084
    5 IF(ISV(LL))3,3,86                                                 DATAR085
    3 LI(LL)=KNALB                                                      DATAR086
      GO TO 86                                                          DATAR087
   85 NFLAG = NFLAG+1                                                   DATAR088
      NDA(LL)=IDKFO(ILB)                                                DATAR089
      IZ=IZ+1                                                           DATAR090
   86 CONTINUE                                                          DATAR091
      IF(IZ) 88,88,87                                                   DATAR092
   87 IDT = IFAIL                                                       DATAR093
   88 PRINT703,J,JB,MJ(J),(ISV(LL),LI(LL),NDA(LL),LL=1,3),IDT           DATAR094
  703 FORMAT(9H0EQUATION,I4,1X7HOF TYPE,1XI2,1XA8,                      DATAR095
     13(3X,1H(,I2,1H,,2A4,1H),2X) ,  1X,A4)                             DATAR096
    4 CONTINUE                                                          DATAR097
      RETURN                                                            DATAR098
      END                                                               DATAR099
```

```
FORTRAN IV G LEVEL  20              DRIGHT             DATE = 72258         14/03/41                PAGE 0001

 0001                SUBROUTINE DRIGHT( FOR)                                              DRIGH000
 0002                DIMENSION FOR(1)                                                     DRIGH001
 0003                REAL*8 IDJOB,IDATE,ID,IDJ                                            DRIGH002
 0004                COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100),    DRIGH003
                    1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),       DRIGH004
                    2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),KFO(100),        DRIGH005
                    3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100),    DRIGH006
                    4JNO(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),DRIGH007
                    5PCEL(100),JBV(100),QD(100),DD(100),BDD(100),                        DRIGH008
                    6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM     DRIGH009
 0005                DO 102  I=1,JBM                                                      DRIGH010
 0006            102 FOR(I)=0.0                                                           DRIGH011
 0007                DO 991J=1,JMAX                                                       DRIGH012
 0008                JB=JBV(J)                                                            DRIGH013
 0009                I =ISA(J)                                                            DRIGH014
 0010                X=XSA(J)                                                             DRIGH015
 0011                K =ISB(J)                                                            DRIGH016
 0012                Y=XSB(J)                                                             DRIGH017
 0013                KK =ISC(J)                                                           DRIGH018
 0014                Z=XSC(J)                                                             DRIGH019
 0015                GO TO (1,1,1,4,5,5,5,5,5,10,11,12,13,14) , JB                        DRIGH020
 0016              1 FOR(J)=YD(K,Y) -YD( I,X)                                             DRIGH021
 0017                GO TO 99                                                             DRIGH022
 0018              4 FOR(J)=-YD(I,X)                                                      DRIGH023
 0019                GO TO 99                                                             DRIGH024
 0020              5 FOR(J)=Q(I)*YD(I,X)-           YDP(I,X)+           YDP(K,Y)          DRIGH025
                    1- Q(K)*YD(K,Y)                                                       DRIGH026
 0021                IF (JB-9)99,8,9                                                      DRIGH027
 0022              8 FOR(J)=FOR(J)+            YDP(KK,Z) - Q(KK)*YD (KK,Z)                DRIGH028
 0023                GO TO99                                                              DRIGH029
 0024              9 FOR(J)=FOR(J)+Q(KK)*YD(KK,Z)-            YDP(KK,Z)                   DRIGH030
 0025                GO TO 99                                                             DRIGH031
 0026             10 FOR(J)=           YDP(K,Y)-Q(K)*YD(K,Y)+QD(J)*DD(J)                  DRIGH032
 0027                GO TO 99                                                             DRIGH033
 0028             11 FOR(J)=YDP(I,X)-Q(I)*YD(I,X)+FF(K)*DK(K)*B(K)*VOL(K)                 DRIGH034
 0029                GO TO 99                                                             DRIGH035
 0030             12 FOR(J)=YDP(I,X)-Q(I)*YD(I,X)+FF(KK)*DK(KK)*B(KK)*VOL(KK)             DRIGH036
 0031                FOR(J)=FOR(J)+YDP(K,Y)-Q(I)*YD(K,Y)                                  DRIGH037
 0032                GO TO 99                                                             DRIGH038
 0033             13 FOR(J)=Q(I)*YD(I,X)-YDP(I,X)+FF(K)*DK(K)*B(K)*VOL(K)                 DRIGH039
 0034                GO TO 99                                                             DRIGH040
 0035             14 FOR(J)=Q(I)*YD(I,X)-YDP(I,X)+FF(KK)*DK(KK)*B(KK)*VOL(KK)             DRIGH041
 0036                FOR(J)=FOR(J)+Q(K)*YD(K,Y)-YDP(K,Y)                                  DRIGH042
 0037             99 CONTINUE                                                             DRIGH043
 0038            991 CONTINUE                                                             DRIGH044
 0039                RETURN                                                               DRIGH045
 0040                END                                                                  DRIGH046
```

```
FORTRAN IV G LEVEL  20              FUN              DATE = 72258          14/03/41                PAGE 0001

 0001              FUNCTION FUN(I,X)                                                     FUN  000
 0002              REAL*8 IDJOB,IDATE,ID,IDJ                                             FUN  001
 0003              COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100),   FUN  002
                  1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),       FUN  003
                  2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),KFO(100),        FUN  004
                  3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100),    FUN  005
                  4JND(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),FUN  006
                  5PCEL(100),JBV(100),GD(100),DD(100),BDD(100),                        FUN  007
                  6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM     FUN  008
 0004              ENTRY FR  (I,X)                                                       FUN  009
 0005              FUN=F(S,V,I,X)                                                        FUN  010
 0006              FR   = FUN                                                            FUN  011
 0007              RETURN                                                                FUN  012
 0008              ENTRY FA  (I,X)                                                       FUN  013
 0009              FUN=F(SD,VD,I,X)                                                      FUN  014
 0010              FA   = FUN                                                            FUN  015
 0011              RETURN                                                                FUN  016
 0012              ENTRY FBR (I,X)                                                       FUN  017
 0013              FUN=FB(S,V,I,X)                                                       FUN  018
 0014              FBR  = FUN                                                            FUN  019
 0015              RETURN                                                                FUN  020
 0016              ENTRY FBA (I,X)                                                       FUN  021
 0017              FUN=FB(SD,VD,I,X)                                                     FUN  022
 0018              FBA  = FUN                                                            FUN  023
 0019              RETURN                                                                FUN  024
 0020              ENTRY FPR (I,X)                                                       FUN  025
 0021              FUN = FP(S,V,I,X)                                                     FUN  026
 0022              FPR  = FUN                                                            FUN  027
 0023              RETURN                                                                FUN  028
 0024              ENTRY FPA (I,X)                                                       FUN  029
 0025              FUN = FP(SD,VD,I,X)                                                   FUN  030
 0026              FPA  = FUN                                                            FUN  031
 0027              RETURN                                                                FUN  032
 0028              ENTRY FBPR(I,X)                                                       FUN  033
 0029              FUN = FBP(S,V,I,X)                                                    FUN  034
 0030              FBPR = FUN                                                            FUN  035
 0031              RETURN                                                                FUN  036
 0032              ENTRY FBPA(I,X)                                                       FUN  037
 0033              FUN = FBP(SD,VD,I,X)                                                  FUN  038
 0034              FBPA = FUN                                                            FUN  039
 0035              RETURN                                                                FUN  040
 0036              ENTRY YBP (I,X)                                                       FUN  041
 0037              FUN = YBPT(S,V,I,X)                                                   FUN  042
 0038              YBP  = FUN                                                            FUN  043
 0039              RETURN                                                                FUN  044
 0040              END                                                                   FUN  045
```

```
FORTRAN IV G LEVEL  20                MAMS                DATE = 72258         14/03/41             PAGE 0001

  0001                SUBROUTINE MAMS ( A,B,C,L1,L2,L3,NRDIM)                         MAMS 000
  0002                DIMENSION A(NRDIM,1),B(NRDIM,1),C(NRDIM,1)                      MAMS 001
  0003                DO 2  I=1,L1                                                    MAMS 002
  0004                DO 2J=1,L3                                                      MAMS 003
  0005                S=0.0                                                           MAMS 004
  0006                DO 1  K=1,L2                                                    MAMS 005
  0007              1 S=S+A(I,K) * B(K,J)                                             MAMS 006
  0008              2 C(I,J) = S                                                      MAMS 007
  0009                RETURN                                                          MAMS 008
  0010                END                                                             MAMS 009
```

```
FORTRAN IV G LEVEL  20              MATINV             DATE = 72258         14/03/41                PAGE 0001

      0001                   SUBROUTINE MATINV (A,N,B,M,DETERM,JMAX,IND,IKND)                  MATIN000
      0002                   DIMENSION IPIVOT(100),A(100,1),B(100,1),INDEX(100,2),PIVOT(100)   MATIN001
      0003                   IF(IND)10,10,1003                                                 MATIN002
      0004              1003 GO TO(1004,1006),IKND                                             MATIN003
      0005              1004 PRINT 1005                                                        MATIN004
      0006              1005 FORMAT('1',51X,'INITIAL BOD MATRIX'                               MATIN005
      0007                   GO TO 1008                                                        MATIN006
      0008              1006 PRINT 1007                                                        MATIN007
      0009              1007 FORMAT('1',48X,'INITIAL DO DEFICIT MATRIX'                        MATIN008
      0010              1008 PRINT 1000,(J,J=1,JMAX)                                           MATIN009
      0011              1000 FORMAT(   /4X,9I13/)                                              MATIN010
      0012                   DO 1001 I=1,JMAX                                                  MATIN011
      0013              1001 PRINT 2000,I,(A(I,J),J=1,JMAX)                                    MATIN012
      0014                   PRINT 1009                                                        MATIN013
      0015              1009 FORMAT(/' FORCING FUNCTIONS'/)                                    MATIN014
      0016                   PRINT 2000,I,(B(J,1),J=1,JMAX)                                    MATIN015
      0017              2000 FORMAT(I3,1X,9E13.5/(4X,9E13.5))                                  MATIN016
      0018                10 DETERM=1.0                                                        MATIN017
      0019                15 DO 20 J=1,N                                                       MATIN018
      0020                20 IPIVOT(J)=0                                                       MATIN019
      0021                30 DO 550 I=1,N                                                      MATIN020
                    C                                                                          MATIN021
                    C        SEARCH FOR PIVOT ELEMENT                                          MATIN022
                    C                                                                          MATIN023
      0022                40 AMAX=0.0                                                          MATIN024
      0023                45 DO 105 J=1,N                                                      MATIN025
      0024                50 IF (IPIVOT(J)-1) 60, 105, 60                                      MATIN026
      0025                60 DO 100 K=1,N                                                      MATIN027
      0026                70 IF (IPIVOT(K)-1) 80, 100, 740                                     MATIN028
      0027                80 IF (ABS(AMAX)-ABS(A(J,K))) 85, 100, 100                           MATIN029
      0028                85 IROW=J                                                            MATIN030
      0029                90 ICOLUM=K                                                          MATIN031
      0030                95 AMAX=A(J,K)                                                       MATIN032
      0031               100 CONTINUE                                                          MATIN033
      0032               105 CONTINUE                                                          MATIN034
      0033               110 IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1                                   MATIN035
                    C                                                                          MATIN036
                    C        INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL                 MATIN037
                    C                                                                          MATIN038
      0034               130 IF (IROW-ICOLUM) 140, 260, 140                                    MATIN039
      0035               140 DETERM=-DETERM                                                    MATIN040
      0036               150 DO 200 L=1,N                                                      MATIN041
      0037               160 SWAP=A(IROW,L)                                                    MATIN042
      0038               170 A(IROW,L)=A(ICOLUM,L)                                             MATIN043
      0039               200 A(ICOLUM,L)=SWAP                                                  MATIN044
      0040               205 IF(M) 260, 260, 210                                               MATIN045
      0041               210 DO 250 L=1, M                                                     MATIN046
      0042               220 SWAP=B(IROW,L)                                                    MATIN047
      0043               230 B IROW,L)=B(ICOLUM,L)                                             MATIN048
      0044               250 B ICOLUM,L)=SWAP                                                  MATIN049
      0045               260 INDEX(I,1)=IROW                                                   MATIN050
      0046               270 INDEX(I,2)=ICOLUM                                                 MATIN051
```

```
 0047          310 PIVOT(I)=A(ICOLUM,ICOLUM)                                     MATIN052
 0048          320 DETERM=DETERM*PIVOT(I)                                        MATIN053
             C                                                                   MATIN054
             C      DIVIDE PIVOT ROW BY PIVOT ELEMENT                            MATIN055
             C                                                                   MATIN056
 0049          330 A(ICOLUM,ICOLUM)=1.0                                          MATIN057
 0050          340 DO 350 L=1,N                                                  MATIN058
 0051          350 A(ICOLUM,L)=A(ICOLUM,L)/PIVOT(I)                              MATIN059
 0052          355 IF(M) 380, 380, 360                                           MATIN060
 0053          360 DO 370 L=1,M                                                  MATIN061
 0054          370 B(ICOLUM,L)=B(ICOLUM,L)/PIVOT(I)                              MATIN062
             C                                                                   MATIN063
             C      REDUCE NON-PIVOT ROWS                                        MATIN064
             C                                                                   MATIN065
 0055          380 DO 550 L1=1,N                                                 MATIN066
 0056          390 IF(L1-ICOLUM) 400, 550, 400                                   MATIN067
 0057          400 T=A(L1,ICOLUM)                                                MATIN068
 0058          420 A(L1,ICOLUM)=0.0                                              MATIN069
 0059          430 DO 450 L=1,N                                                  MATIN070
 0060          450 A(L1,L)=A(L1,L)-A(ICOLUM,L)*T                                 MATIN071
 0061          455 IF(M) 550, 550, 460                                           MATIN072
 0062          460 DO 500 L=1,M                                                  MATIN073
 0063          500 B(L1,L)=B(L1,L)-B(ICOLUM,L)*T                                 MATIN074
 0064          550 CONTINUE                                                      MATIN075
             C                                                                   MATIN076
             C      INTERCHANGE COLUMNS                                          MATIN077
             C                                                                   MATIN078
 0065          600 DO 710 I=1,N                                                  MATIN079
 0066          610 L=N+1-I                                                       MATIN080
 0067          620 IF (INDEX(L,1)-INDEX(L,2)) 630, 710, 630                      MATIN081
 0068          630 JROW=INDEX(L,1)                                               MATIN082
 0069          640 JCOLUM=INDEX(L,2)                                             MATIN083
 0070          650 DO 705 K=1,N                                                  MATIN084
 0071          660 SWAP=A(K,JROW)                                                MATIN085
 0072          670 A(K,JROW)=A(K,JCOLUM)                                         MATIN086
 0073          700 A(K,JCOLUM)=SWAP                                              MATIN087
 0074          705 CONTINUE                                                      MATIN088
 0075          710 CONTINUE                                                      MATIN089
 0076          740 CONTINUE                                                      MATIN090
 0077          750 CONTINUE                                                      MATIN091
 0078              IF(IND)1020,1020,1010                                         MATIN092
 0079         1010 GO TO(1011,1014),IKNO                                         MATIN093
 0080         1011 PRINT 1012                                                    MATIN094
 0081         1012 FORMAT(///50X,'INVERTED BOD MATRIX'                           MATIN095
 0082              GO TO 1016                                                    MATIN096
 0083         1014 PRINT 1015                                                    MATIN097
 0084         1015 FORMAT(///47X,'INVERTED DO DEFICIT MATRIX'                    MATIN098
 0085         1016 PRINT 1000,(J,J=1,JMAX)                                       MATIN099
 0086              DO 1002 I=1,JMAX                                              MATIN100
 0087         1002 PRINT 2000,I,(A(I,J),J=1,JMAX)                                MATIN101
 0088              PRINT 1017                                                    MATIN102
 0089         1017 FORMAT(/' SOLUTIONS FOR UNKNOWN COEFFICIENTS'/)               MATIN103
```

```
FORTRAN IV G LEVEL  20              MATINV          DATE = 72258        14/03/41            PAGE 0003

  0090                PRINT 2000,I,(B(J,I),J=1,JMAX)                                MATIN104
  0091          1020 RETURN                                                        MATIN105
  0092                END                                                          MATIN106
```

```
FORTRAN IV G LEVEL  20                MATS                DATE = 72258           14/03/41                PAGE 0001

  0001              SUBROUTINE MATS(F,FB,FP,FBP,IJK,TM)                                         MATS 000
  0002              REAL*8 IDJOB,IDATE,ID,IDJ                                                   MATS 001
  0003              COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100),           MATS 002
                   1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),               MATS 003
                   2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),KFD(100),                MATS 004
                   3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100),            MATS 005
                   4JNO(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),          MATS 006
                   5PDEL(100),JBV(100),QD(100),DD(100),BDD(100),                                MATS 007
                   6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM             MATS 008
  0004              EXTERNAL F,FB,FP,FBP                                                        MATS 009
  0005              DIMENSION TM(100,1)                                                         MATS 010
  0006              DO 101 I=1,JBM                                                              MATS 011
  0007              DO 101 J=1,JBM                                                              MATS 012
  0008          101 TM(I,J)=0.0                                                                 MATS 013
  0009              DO199 J=1,JMAX                                                              MATS 014
  0010              JB=JBV(J)                                                                   MATS 015
  0011              I =ISA(J)                                                                   MATS 016
  0012              X=XSA(J)                                                                    MATS 017
  0013              L=NOSECT(I ,IS2)                                                            MATS 018
  0014              IM=L-1                                                                      MATS 019
  0015              K =ISB(J)                                                                   MATS 020
  0016              Y=XSB(J)                                                                    MATS 021
  0017              M=NOSECT(K ,IS2)                                                            MATS 022
  0018              KM=M-1                                                                      MATS 023
  0019              KK =ISC(J)                                                                  MATS 024
  0020              Z=XSC(J)                                                                    MATS 025
  0021              N=NOSECT(KK ,IS2)                                                           MATS 026
  0022              KKM= N-1                                                                    MATS 027
  0023              GOTO(1,2,3,4,5,6,7,8,9,10,11,12,13,14),JB                                   MATS 028
  0024            1 TM(IM,J)=F(I,X)                                                             MATS 029
  0025              TM(L,J)=FB(I,X)                                                             MATS 030
  0026              TM(KM,J)=-F(K,Y)                                                            MATS 031
  0027              TM(M,J)=-FB(K,Y)                                                            MATS 032
  0028              GO TO  99                                                                   MATS 033
  0029            2 TM(L,J)=F(I,X)                                                              MATS 034
  0030              TM(KM,J)=-F(K,Y)                                                            MATS 035
  0031              TM(M,J)=-FB(K,Y)                                                            MATS 036
  0032              GO TO  99                                                                   MATS 037
  0033            3 TM(IM,J)=F(I,X)                                                             MATS 038
  0034              TM(L,J)=FB(I,X)                                                             MATS 039
  0035              TM(M,J)=-FB(K,Y)                                                            MATS 040
  0036              GO TO  99                                                                   MATS 041
  0037            4 TM(IM,J)=F(I,X)                                                             MATS 042
  0038              TM(L,J)=FB(I,X)                                                             MATS 043
  0039              TM(M,J)=-1.0                                                                MATS 044
  0040              GO TO 99                                                                    MATS 045
  0041            5 TM(IM,J)=FP(I,X)-Q(I)*F(I,X)                                                MATS 046
  0042              TM(L,J)=FBP(I,X)-Q(I)*FB(I,X)                                               MATS 047
  0043              TM(KM,J)=Q(K)*F(K,Y)-FP(K,Y)                                                MATS 048
  0044              TM(M,J)=Q(K)*FB(K,Y)-FBP(K,Y)                                               MATS 049
  0045              GO TO 99                                                                    MATS 050
  0046            6 TM(L ,J)=F P(I,X)-Q(I)*F (I,X)                                              MATS 051
```

```
0047            TM(KM,J)=Q(K)*F(K,Y)-FP(K,Y)                                        MATS 052
0048            TM(M,J)=Q(K)*FB(K,Y)-FBP(K,Y)                                       MATS 053
0049            GC TO 99                                                            MATS 054
0050          7 TM(IM,J)=FP(I,X)-Q(I)*F(I,X)                                        MATS 055
0051            TM(L,J)=FBP(I,X)-Q(I)*FB(I,X)                                       MATS 056
0052            TM(M ,J)=Q(K)*FB(K,Y)-FBP(K,Y)                                      MATS 057
0053            GC TO 99                                                            MATS 058
0054          8 TM(IM,J)=FP(I,X)-Q(I)*F(I,X)                                        MATS 059
0055            TM(L,J)=FBP(I,X)-Q(I)*FB(I,X)                                       MATS 060
0056            TM(KM,J)=Q(K)*F(K,Y)-FP(K,Y)                                        MATS 061
0057            TM(M,J)=Q(K)*FB(K,Y)-FBP(K,Y)                                       MATS 062
0058         28 TM( N,J) = Q(KK)*FB(KK,Z)-FBP(KK,Z)                                 MATS 063
0059            TM(KKM,J)=Q(KK)*F(KK,Z)-              FP(KK,Z)                      MATS 064
0060            GC TO 99                                                            MATS 065
0061          9 TM(IM,J)=FP(I,X)-Q(I)*F(I,X)                                        MATS 066
0062            TM(L,J)=FBP(I,X)-Q(I)*FB(I,X)                                       MATS 067
0063            TM(KM,J)=Q(K)*F(K,Y)-FP(K,Y)                                        MATS 068
0064            TM(M,J)=Q(K)*FB(K,Y)-FBP(K,Y)                                       MATS 069
0065         29 TM(KKM,J)=             FP(KK,Z)-Q(KK)*F (KK,Z)                      MATS 070
0066            TM( N,J)=             FBP(KK,Z)-Q(KK)*FB (KK,Z)                     MATS 071
0067            GC TO 99                                                            MATS 072
0068         10 TM(KM,J)=Q(K)*F(K,Y)-           FP(K,Y)                             MATS 073
0069            TM(M,J)=Q(K)*FB(K,Y)-           FBP(K,Y)                            MATS 074
0070            GO TO 99                                                            MATS 075
0071         11 TM(IM,J) = Q(I)*F(I,X)-           FP(I,X)                           MATS 076
0072            TM(L,J)=Q(I)*FB(I,X)-           FBP(I,X)                            MATS 077
0073            JBB=JB-10                                                           MATS 078
0074            GC TO(15,16),IJK                                                    MATS 079
0075         15 TM(M,J)=VOL(K)*RK(K)                                                MATS 080
0076            GC TO 99                                                            MATS 081
0077         16 TM(M,J)=VOL(K)*AK(K)                                                MATS 082
0078            GC TO 99                                                            MATS 083
0079         12 TM(IM,J)=Q(I)*F(I,X)-FP(I,X)                                        MATS 084
0080            TM(L,J)=Q(I)*FB(I,X) - FBP(I,X)                                     MATS 085
0081            TM(KM,J)=Q(K)*F(K,Y)-           FP(K,Y)                             MATS 086
0082            TM(M,J)=Q(K)*FB(K,Y)-           FBP(K,Y)                            MATS 087
0083            GC TO(17,18),IJK                                                    MATS 088
0084         17 TM( N,J)=VOL(KK)*RK(KK)                                             MATS 089
0085            GC TO 99                                                            MATS 090
0086         18 TM( N,J)=VOL(KK)*AK(KK)                                             MATS 091
0087            GO TO 99                                                            MATS 092
0088         13 TM(IM,J)=          FP(I,X)-Q(I)*F(I,X)                              MATS 093
0089            TM(L,J)=          FBP(I,X)-Q(I)*FB(I,X)                             MATS 094
0090            JBB=JB-12                                                           MATS 095
0091            GC TO(19,20),IJK                                                    MATS 096
0092         19 TM(M,J)=Q(K)+VOL(K)*(TIDE1(K)+RK(K))                                MATS 097
0093            GC TO 99                                                            MATS 098
0094         20 TM(M,J)=Q(K)+VOL(K)*(TIDE1(K)+AK(K))                                MATS 099
0095            GC TO 99                                                            MATS 100
0096         14 TM(IM,J) = FP(I,X)-Q(I) *F(I,X)                                     MATS 101
0097            TM(L,J)= FBP(I,X)-Q(I)*FB(I,X)                                      MATS 102
0098            TM(KM,J)=          FP(K,Y)-Q(K)*F (K,Y                              MATS 103
0099               TM(M,J)=        FBP(K,Y)-Q(K)*FB(K,Y)                          MATS 104
0100               GC TO(21,22),IJK                                               MATS 105
0101            21 TM( N,J) = Q(KK)+VOL(KK)*(TIDE1(KK)+RK(KK))                    MATS 106
0102               GO TO 99                                                       MATS 107
0103            22 TM( N,J) = Q(KK)+VOL(KK)*(TIDE1(KK)+AK(KK))                    MATS 108
0104            99 CCNTINUE                                                       MATS 109
0105           199 CONTINUE                                                       MATS 110
0106               RETURN                                                         MATS 111
0107               END                                                            MATS 112
```

```
FORTRAN IV G LEVEL  20            NOSECT            DATE = 72.58          14/03/41            PAGE 0001

    0001              FUNCTION NOSECT (IL,IS2)                                      NOSEC000
    0002              IF(IL-IS2) 1,1,2                                              NOSEC001
    0003            1 NOSECT=2*IL                                                   NOSEC002
    0004              RETURN                                                        NOSEC003
    0005            2 NOSECT=2*IS2+(IL-IS2)                                         NOSEC004
    0006              RETURN                                                        NOSEC005
    0007              END                                                           NOSEC006
```

```
0001          FUNCTION NTPS (I)                                                        NTPS 000
0002          REAL*8 IDJCB,IDATE,ID,IDJ                                                NTPS 001
0003          COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100),        NTPS 002
             1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),            NTPS 003
             2S(100),JMAX,IS1,ISC(100),CK(100),NJU(100),XJU(100),KFD(100),             NTPS 004
             3V(100),SD(100),P(100),G(100),AREA(100),TEMP(100),E(100),HT(100),         NTPS 005
             4JNO(100),W(100),VD(100),T(100),H(100),VOL(100),TIDE1(100),PR(100),NTPS 006
             5PCEL(100),JBV(100),QD(100),DD(100),BDD(100),                             NTPS 007
             6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM          NTPS 008
0004          IF(I) 1,1,2                                                              NTPS 009
0005        1 NTPS =5                                                                  NTPS 010
0006          RETURN                                                                   NTPS 011
0007        2 IF(I-IS2) 4,4,7                                                          NTPS 012
0008        4 IF(KFD(I)-1) 5,6,5                                                       NTPS 013
0009        5 NTPS =6                                                                  NTPS 014
0010          RETURN                                                                   NTPS 015
0011        6 NTPS =1                                                                  NTPS 016
0012          RETURN                                                                   NTPS 017
0013        7 IF(I-J MAX) 8,8,5                                                        NTPS 018
0014        8 K=KFD(I)                                                                 NTPS 019
0015          IF(K.GT.1.AND.K.LE.4) GO TO 9                                            NTPS 020
0016          GO TO 5                                                                  NTPS 021
0017        9 NTPS =K                                                                  NTPS 022
0018          RETURN                                                                   NTPS 023
0019          END                                                                      NTPS 024
```

```
FORTRAN IV G LEVEL  20                 OM                  DATE = 72258         14/03/41               PAGE 0001

0001                    FUNCTION OM(KFO,BG,CH,F,FB,I,X)                                    OM    000
0002                    DIMENSION KFO(1),BG(1),CH(1)                                       OM    001
0003                    EXTERNAL  F,FB                                                     OM    002
0004                    M=KFO(I)                                                           OM    003
0005                    GO TO (1,2,3,4),M                                                  OM    004
0006                  1 OM=BG(I)*F(I,X)+CH(I)*FB(I,X)                                      OM    005
0007                    GO TO 5                                                            OM    006
0008                  3 OM=               CH(I)*FB(I,X)                                    OM    007
0009                    GO TO 5                                                            OM    008
0010                  2 OM=BG(I)*F(I,X)                                                    OM    009
0011                    GO TO 5                                                            OM    010
0012                  4 OM= BG(I)                                                          OM    011
0013                5   CONTINUE                                                           OM    012
0014                    RETURN                                                             OM    013
0015                    END                                                                OM    014
```

```
FORTRAN IV G LEVEL  20              PFUN                DATE = 72258         14/03/41            PAGE 0001

                                                                              PFUN 000
 0001              FUNCTION PFUN(SSD,VVD,I,X)                                 PFUN 001
 0002              DIMENSION SSD(1),VVD(1)                                    PFUN 002
 0003              REAL*8 IDJOB,IDATE,ID,IDJ                                  PFUN 003
 0004              COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100), PFUN 004
                  1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100), PFUN 005
                  2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),XFO(100), PFUN 006
                  3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100), PFUN 007
                  4JND(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),PFUN 008
                  5PEEL(100),JBV(100),QD(100),DD(100),BDD(100),               PFUN 009
                  6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM PFUN 010
 0005              ENTRY F   (SSD,VVD,I,X)                                    PFUN 011
 0006              K=1                                                        PFUN 012
 0007              GOTO9                                                      PFUN 013
 0008              ENTRY FB  (SSD,VVD,I,X)                                    PFUN 014
 0009              K=2                                                        PFUN 015
 0010              GO TO 9                                                    PFUN 016
 0011              ENTRY YBT (SSD,VVD,I,X)                                    PFUN 017
 0012              K=3                                                        PFUN 018
 0013              GOTO 9                                                     PFUN 019
 0014              ENTRY FP  (SSD,VVD,I,X)                                    PFUN 020
 0015              K=4                                                        PFUN 021
 0016              GO TO 9                                                    PFUN 022
 0017              ENTRY FBP (SSD,VVD,I,X)                                    PFUN 023
 0018              K=5                                                        PFUN 024
 0019              GO TO 9                                                    PFUN 025
 0020              ENTRY YBPT(SSD,VVD,I,X)                                    PFUN 026
 0021              K=6                                                        PFUN 027
 0022              GO TO 9                                                    PFUN 028
 0023            9 GO TO (11,12,13,14,15,16) ,K                               PFUN 029
 0024           11 PFUN = EXP (X*SSD(I))                                      PFUN 030
 0025              F    = PFUN                                                PFUN 031
 0026              RETURN                                                     PFUN 032
 0027           12 PFUN = EXP (X*VVD(I))                                      PFUN 033
 0028              FB   = PFUN                                                PFUN 034
 0029              RETURN                                                     PFUN 035
 0030           13 PFUN = WU(I)/(AREA(I)*RK(I))                               PFUN 036
 0031              YBT  = PFUN                                                PFUN 037
 0032              RETURN                                                     PFUN 038
 0033           14 PFUN=SSD(I)*EXP (X*SSD(I))*E(I)*AREA(I)                    PFUN 039
 0034              FP   = PFUN                                                PFUN 040
 0035              RETURN                                                     PFUN 041
 0036           15 PFUN = VVD(I)*EXP (X*VVD(I))*E(I)*AREA(I)                  PFUN 042
 0037              FBP  = PFUN                                                PFUN 043
 0038              RETURN                                                     PFUN 044
 0039           16 PFUN = 0.0                                                 PFUN 045
 0040              YBPT = PFUN                                                PFUN 046
 0041              RETURN                                                     PFUN 047
 0042              END
```

```
FORTRAN IV G LEVEL  20              PLACE            DATE = 72258         14/03/41            PAGE 0001

  0001                SUBROUTINE PLACE(J.JJ.IS2M.ISA.LINE)                          PLACE000
  0002                DIMENSION LINE(41).ISA(1)                                     PLACE001
  0003                IS2 = IS2M/2                                                  PLACE002
  0004                KS=ISA(JJ)                                                    PLACE003
  0005                IF(KS) 4.4.1                                                  PLACE004
  0006              1 IS=2*KS                                                       PLACE005
  0007                IF(IS-IS2M) 2.2.3                                             PLACE006
  0008              2 LINE(IS)=J                                                    PLACE007
  0009                LINE(IS-1)=J                                                  PLACE008
  0010                GC TO 4                                                       PLACE009
  0011              3 IS=IS2+KS                                                     PLACE010
  0012                LINE(IS)=J                                                    PLACE011
  0013              4 RETURN                                                        PLACE012
  0014                END                                                           PLACE013
```

```
FORTRAN IV G LEVEL  20                PRI                DATE = 72258          14/03/41                    PAGE 0001

 0001               SUBROUTINE PRI(MI,LJ,NK)                                                    PRI   000
 0002               REAL*8 IDJOB,IDATE,ID,IDJ                                                   PRI   001
 0003               COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),PD(100),           PRI   002
                   1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),              PRI   003
                   2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),KFD(100),               PRI   004
                   3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100),           PRI   005
                   4JNO(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),PRI      006
                   5PDEL(100),JBV(100),CD(100),DD(100),BOD(100),                               PRI   007
                   6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM            PRI   008
 0004               COMMON/NEW/FAC1,FAC2,FAC3,ITYPE(35)                                         PRI   009
 0005               DIMENSION LJ(1),MI(1)                                                       PRI   010
 0006               PRINT 10                                                                    PRI   011
 0007               DO 1 I=1,IMAX                                                               PRI   012
 0008            10 FORMAT(///5X7HSECTION,4X,    4H KFD,2X,'JUNCTION ID AND DISTANCE',PRI       013
                   112X,2HRK,8X2HAK,8X2HDK,9X1HE,8X2HFF,6X,5HTEMP1X,'PDEL'/28X,'AT',           PRI   014
                   2' ENDS',20X,'REM.RATE ',' REA.RATE ',' DEO.RATE ',' DISP.RATE ',           PRI   015
                   3' ULT/5DAY ',4X,' MILE-INCR'/56X,'(1/DAY)',3X,'(1/DAY)',3X,                PRI   016
                   4'(1/DAY)',2X,'(MI2/DAY)',11X,'(CENT)',2X,'(MI)'/)                          PRI   017
 0009             1 PRINT 20,I,ID(I),MI(I),KFD(I),NJD(I),XJD(I),NJU(I),                         PRI   018
                   1XJU(I),RK(I),AK(I),DK(I),E(I), FF(I),TEMP(I),PDEL(I)                       PRI   019
 0010            20 FORMAT(1H0,I2,1XA4,1XA4,3X,I4,1X,2(1X,1H(,A3,1H,F6.1,1H),2X),               PRI   020
                   16F10.3,F7.3)                                                               PRI   021
 0011               IF(NK)31,31,32                                                              PRI   022
 0012            31 PRINT 30                                                                    PRI   023
 0013            30 FORMAT(///5X7HSECTION,6X5HCROSS,10X1HQ,13X2HBD,12X2HPR,10X5HTIDE1,PRI       024
                   111X,2HHT,8X,3HVOL,12X,'WU'/17X,'SEC-AREA',6X,'FLOW',7X,'BENTHL-DMDPRI      025
                   2',5X'PHOTO-RESP',4X'TIDE-COEF',7X'DEPTH',5X'VOLUME',8X,'UNF-LOAD' PRI      026
 0014               GO TO 34                                                                    PRI   027
 0015            32 PRINT 33                                                                    PRI   028
 0016            33 FORMAT(///5X7HSECTION,6X5HCROSS,       10X1HQ,13X2HBD,12X2HPR,              PRI   029
                   110X5HTIDE1,11X,2HHT,8X,3HVOL,8X,'WU/AREA*RK'/17X,'SEC-AREA',6X,            PRI   030
                   2                'FLOW',7X,'BENTHL-DMD',5X,'PHOTO-RESP',4X,'TIDE-COEF'PRI   031
                   3,7X,'DEPTH',5X,'VOLUME',8X,'UNF-LOAD'                                      PRI   032
 0017            34 IF(NK)5,5,7                                                                 PRI   033
 0018             5 PRINT 6                                                                     PRI   034
 0019             6 FORMAT(17X,'(FT**2)',7X,            '(CFS)',5X,'(GMS/M2-DAY)',              PRI   035
                   14X,'(MG/L-DAY)',20X,'(FT.)',4X,'(FT**3)',6X,'(LBS/DAY-MI)'                 PRI   036
 0020               GO TO 9                                                                     PRI   037
 0021             7 PRINT 8                                                                     PRI   038
 0022             8 FORMAT(17X,'(MI**2)',4X,           '(MI**3/DAY)',2X,                        PRI   039
                   1'(GMS/M2-DAY)',4X,'(MG/L-DAY)',20X,'(MTRS)',4X,'(MI**3)',8X,               PRI   040
                   2'(MG/L)'                                                                   PRI   041
 0023             9 DO 2 I=1,IMAX                                                               PRI   042
 0024             2 PRINT 40,I,ID(I),MI(I),AREA(I),Q(I),BD(I),PR(I),TIDE1(I),                   PRI   043
                   1HT(I),VOL(I),WU(I)                                                         PRI   044
 0025               RETURN                                                                      PRI   045
 0026            40 FORMAT(1H0I2,1XA4,1XA4,2E12.3,5X,E12.3,2X,E12.3,1X,E12.3,1X,                PRI   046
                   12E12.3,3X,E12.3)                                                           PRI   047
 0027               ENTRY PRII  (MI,LJ,NK)                                                      PRI   048
 0028               PRINT 50                                                                    PRI   049
 0029               DO 3 I=1,IMAX                                                               PRI   050
 0030            50 FORMAT(///3X7HSECTION,8X,1HU,9X1HS,9X1HV,8X2HSD,                            PRI   051
                 18X2HVD,9X1HB,9X1HC,9X1HG,9X1HH,8X2HZZ5X3HFAC/)                 PRI  052
  0031         3 PRINT 60,I,ID(I),MI(I),U(I),S(I),V(I),SD(I),VD(I),B(I),C(I) ,   PRI  053
                1G(I),H(I),ZZ(I),FAC(I)                                          PRI  054
  0032           RETURN                                                          PRI  055
  0033        60 FORMAT(1H0,I2,1XA4,1XA4,9F10.5,2F8.2)                           PRI  056
  0034           ENTRY PRJ   (MI,LJ,NK)                                          PRI  057
  0035           PRINT 70                                                        PRI  058
  0036        70 FORMAT(/1H05X8HEQUATION,4X,'JUNCTION ID, SECTIONS',4X,          PRI  059
                117X1HW,13X2HQD,10X2HDD,9X3HBDD/18X,'MEETING AT ',              PRI  060
                2'JUNCTION,',19X,'PT-LOAD',7X,'FLOW-DAM',5X,'DEF-DAM',4X,'BOD-DAM' PRI  061
  0037           IF(NK)11,11,14                                                  PRI  062
  0038        11 PRINT 12                                                        PRI  063
  0039        12 FORMAT(18X,'AND DISTANCE',26X,'(LBS/DAY)',7X,'(CFS)',7X,'(MG/L)', PRI  064
                15X,'(MG/L)'                                                     PRI  065
  0040           GO TO 16                                                        PRI  066
  0041        14 PRINT 15                                                        PRI  067
  0042        15 FORMAT(18X,'AND DISTANCE',21X,'(MI**3/DAY)(MG/L)',2X,'(MI**3/DAY)'PRI  068
                1,4X,'(MG/L)',5X,'(MG/L)'                                        PRI  069
  0043        16 DO 4 J=1,JMAX                                                   PRI  070
  0044        4  PRINT80,J,JBV(J),LJ(J),JND(J),ISA(J),XSA(J),ISB(J),             PRI  071
                1XSB(J),ISC(J),XSC(J),W(J),QD(J),DD(J),BDD(J)                    PRI  072
  0045        80 FORMAT(1H0,I2,I3,1XA4,1XA3,1X,3(1H(,I3,F5.1,1H)),6XF12.5,3X,    PRI  073
                13F12.5)                                                         PRI  074
  0046           RETURN                                                          PRI  075
  0047           END                                                             PRI  076
```

```
FORTRAN IV G LEVEL  20                SVD              DATE = 72258         14/03/41              PAGE 0001

      0001                SUBROUTINE SVD(U,E,A,S,V)                                 SVD  000
      0002                IF(U) 2,1,2                                               SVD  001
      0003              1 S=SQRT (A/E)                                              SVD  002
      0004                V=-S                                                      SVD  003
      0005                RETURN                                                    SVD  004
      0006              2 SC=SQRT (1.+4.*A*E/U**2)                                  SVD  005
      0007                P=.5*U/E                                                  SVD  006
      0008                S=P*(1.+SQ)                                               SVD  007
      0009                V=P*(1.-SC)                                               SVD  008
      0010                RETURN                                                    SVD  009
      0011                END                                                       SVD  010
      0001              SUBROUTINE SVD(U,E,A,S,V)                                    SVD  000
      0002              IF(U) 2,1,2                                                  SVD  001
      0003            1 S=SQRT (A/E)                                                 SVD  002
      0004              V=-S                                                         SVD  003
      0005              RETURN                                                       SVD  004
      0006            2 SC=SQRT (1.+4.*A*E/U**2)                                     SVD  005
      0007              P=.5*U/E                                                     SVD  006
      0008              S=P*(1.+SQ)                                                  SVD  007
      0009              V=P*(1.-SC)                                                  SVD  008
      0010              RETURN                                                       SVD  009
      0011              END                                                          SVD  010
```

```
    FORTRAN IV G LEVEL  20                YB                  DATE = 72258          14/03/41                  PAGE 0001

      0001                FUNCTION YB(I,X)                                                       YB    000
      0002                REAL*8 IDJOB,IDATE,ID,IDJ                                              YB    001
      0003                COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100),     YB    002
                         1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),         YB    003
                         2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),KFO(100),          YB    004
                         3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100),      YB    005
                         4JND(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),    YB    006
                         5PDEL(100),JBV(100),GD(100),DD(100),BDD(100),                          YB    007
                         6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM       YB    008
      0004                IF(RK(I))2,2,1                                                         YB    009
      0005              1 YB=WU(I)/(AREA(I)*RK(I))                                               YB    010
      0006                YBB=YB                                                                 YB    011
      0007                GO TO 3                                                                YB    012
      0008              2 YB=0.                                                                  YB    013
      0009              3 RETURN                                                                 YB    014
      0010                END                                                                    YB    015
```

```
FORTRAN IV G LEVEL  20          YD          DATE = 72258        14/03/41          PAGE 0001

 0001           FUNCTION YD(I,X)                                                    YD   000
 0002           REAL*8 IDJOB,IDATE,ID,IDJ                                           YD   001
 0003           COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),Q(100),BD(100),   YD   002
               1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),      YD   003
               2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),KFO(100),       YD   004
               3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100),   YD   005
               4JNO(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),YD   006
               5PDEL(100),JBV(100),QD(100),DD(100),BOD(100),                       YD   007
               6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM    YD   008
 0004           EXTERNAL FR,FBR                                                     YD   009
 0005           YD=FAC(I)*OM(KFO,B,C,FR,FBR,I,X) + 'Z(I)                            YD   010
 0006           RETURN                                                              YD   011
 0007           END                                                                 YD   012
```

```
FORTRAN IV G LEVEL  20                YDP                DATE = 72258          14/03/41                PAGE 0001

0001                FUNCTION YDP(I,X)                                                           YDP  000
0002                REAL*8 IDJOB,IDATE,ID,IDJ                                                   YDP  001
0003                COMMON/HY/ISA(100),XSA(100),IDJOB,AK(100),FF(100),O(100),BD(100),           YDP  002
                   1WU(100),IMAX,IS2,ISB(100),XSB(100),RK(100),NJD(100),XJD(100),              YDP  003
                   2S(100),JMAX,IS1,ISC(100),DK(100),NJU(100),XJU(100),KFO(100),               YDP  004
                   3V(100),SD(100),B(100),G(100),AREA(100),TEMP(100),E(100),HT(100),           YDP  005
                   4JNO(100),W(100),VD(100),C(100),H(100),VOL(100),TIDE1(100),PR(100),         YDP  006
                   5PDEL(100),JHV(100),GD(100),DD(100),BDD(100),                               YDP  007
                   6ID(100),IDJ,FAC(100),NSV(100),U(100),IDATE,ZZ(100),XSC(100),JBM            YDP  008
0004                EXTERNAL FPR,FBPR                                                           YDP  009
0005                YDP=FAC(I)*OM(KFO,B,C,FPR,FBPR,I,X)                                         YDP  010
0006                RETURN                                                                      YDP  011
0007                END                                                                         YDP  012
```

Requirements for ES002

1. IBM 1130

   a. 8K words of core
   b. With a disk drive
   c. 60 sectors of disk space for temporary data files.

2. Variable used to designate the input and output units should conform to the input-output devices at the disposal of the user.

   In this program the variables MX and MY are used for this purpose.

```
      REAL IDJOB,IDATE                                                  ES002000
      COMMON FAC1,FAC2,FAC3,ITYPE(35)                                   ES002001
      DEFINE FILE 1(41,82,U,J5)                                         ES002002
      DEFINE FILE 2(41,82,U,J5)                                         ES002003
      DEFINE FILE 4(250,12,U,J5)                                        ES002004
      DEFINE FILE 5(2,82,U,J5)                                          ES002005
      DEFINE FILE 6(2,82,U,J5)                                          ES002006
      DEFINE FILE 7(2,82,U,J5)                                          ES002007
      DEFINE FILE 8(41,6,U,J5)                                          ES002008
      DEFINE FILE 9(41,82,U,J5)                                         ES002009
      DEFINE FILE 10(1,82,U,J5)                                         ES002010
   40 III=1                                                             ES002011
      III2=7                                                            ES002012
      J=1                                                               ES002013
      LLL=1                                                             ES002014
      JBM=41                                                            ES002015
   15 ICON=-1                                                           ES002016
    1 NFLAG=0                                                           ES002017
      CALL DATA1(NFLAG,NN,IDJOB,IDATE,IMAX,JMAX,IS2,1,I,ICON)           ES002018
      CALL ATP1(NFLAG,IMAX,IS2,JMAX,1)                                  ES002019
      CALL DATA1(NFLAG,NN,IDJOB,IDATE,IMAX,JMAX,IS2,2,I,ICON)           ES002020
      CALL ATP1(NFLAG,LLL,IS2,JMAX,2)                                   ES002021
      IF(NFLAG)11,11,1                                                  ES002022
   11 CALL PRI(1,IMAX,JMAX,IDJOB,IDATE,0)                               ES002023
      CALL PRI(3,IMAX,JMAX,IDJOB,IDATE,0)                               ES002024
      CALL DATA2(IMAX,JMAX,IS2,JBM)                                     ES002025
      CALL PLAC1(IS2,JMAX,JBM)                                          ES002026
      IF(NN)2,2,3                                                       ES002027
    2 J=0                                                               ES002028
      K=0                                                               ES002029
    9 J=J+1                                                             ES002030
      K=K+1                                                             ES002031
      CALL MATS(1,2,4,5,J,IS2,JMAX,J)                                   ES002032
      IF(J-1)6,6,7                                                      ES002033
    6 CALL BRIGH(1,IS2,JMAX)                                            ES002034
      GO TO 8                                                           ES002035
    7 CALL DRIGH(1,IS2,JMAX)                                            ES002036
    8 CALL LST(K,7,I,JMAX)                                              ES002037
      CALL MATNV(J,JMAX,1,J,7)                                          ES002038
   50 K=K+1                                                             ES002039
      CALL LST(K,7,I,JMAX)                                              ES002040
   14 CALL BCGH(IS2,JMAX,J,ICON)                                        ES002041
      IF(ICON-1)12,10,12                                                ES002042
   12 IF(J-1)9,9,10                                                     ES002043
   10 CALL PRI(1,IMAX,JMAX,IDJOB,IDATE,1)                               ES002044
      CALL PRI(2,IMAX,JMAX,IDJOB,IDATE,1)                               ES002045
      CALL PRI(3,IMAX,JMAX,IDJOB,IDATE,1)                               ES002046
   16 CALL BDOUT(IMAX,IDJOB,IDATE,ICON)                                 ES002047
      IF(ICON-4)18,17,18                                                ES002048
   17 ICON=5                                                            ES002049
      GO TO 16                                                          ES002050
   18 IF(ICON-2)15,1,15                                                 ES002051
    3 CALL BRIGH(1,IS2,JMAX)                                            ES002052
      IF(I)4,4,5                                                        ES002053
    5 I=2                                                               ES002054
    4 CALL LST(1,7,I,JMAX)                                              ES002055
      CALL MEMS(JMAX,1,1)                                               ES002056
      CALL LST(2,7,I,JMAX)                                              ES002057
      CALL BCGH(IS2,JMAX,1)                                             ES002058
      CALL DRIGH(1,IS2,JMAX)                                            ES002059
      CALL LST(3,7,I,JMAX)                                              ES002060
      CALL MEMS(JMAX,2,2)                                               ES002061
      CALL LST(4,7,I,JMAX)                                              ES002062
      GO TO 14                                                          ES002063
      END                                                               ES002064
      SUBROUTINE ATP1(NFLAG,IMAX,IS2,JMAX,KK)
      REAL IDKFO(6),LI(3),NDA(3),MJ(41)
      REAL IPASS,IFAIL,IDHJB(14),IDT,IBLNK,KNALB
      DIMENSION IDJB(3,14),ISV(3)
      DIMENSION KFO(1),JBV(1),ISA(1),ISB(1),ISC(1),AAA(27)
      DATA IDKFO/'DBLE','S NI','S PI','S CM','MISS','N CL'/
      DATA IDHJB/'C 2D','C DN','C DP','C DC','M 2D','M DN','M DP',
     1'M3D1','M3D2','M DM','MCM1','MCM2','M1CM','M2CM'/
      DATA IDJB/2*1,5,2,1,5,1,3,5,1,4,5,2*1 ,5,2,1,5,1,3,5,
     26*1,1,1,5,1,4,5,2*1,4,1,4,5,2*1,4/
      DATA IPASS/'PASS'/,IFAIL/'FAIL'/,IBLNK/'    '/
      KNALB=IBLNK
      MX=5
      KKK=1
      GO TO(34,33),KK
   33 READ(5'2)MJ
      J=1
      DO 88 J=1,JMAX
      READ(9'J)K1,G1,KFO,G2,G3,JBV,ISA,ISB,ISC,AAA
      JB=JBV(KKK)
      IZ=0
      DO 86 LL=1,3
      GO TO (81,82,83),LL
   81 IS=ISA(KKK)
      GO TO 84
   82 IS=ISB(KKK)
      GO TO 84
   83 IS=ISC(KKK)
   84 ISV(LL)=IS
      I=IS
      IF(I)37,37,23
   23 READ(9'I)K1,G2,KFO
      GO TO 37
   34 DO 11 I=1,IMAX
      READ(9'I)K1,G1,KFO,G2,G3,JBV,ISA,ISB,ISC,AAA
      IF(KFO(1))5,1,2
   37 IF(I)1,1,2
    1 NTPS =5
      GO TO 10
    2 IF(I-IS2)4,4,7
    4 IF(KFO(KKK)-1)5,6,5
    5 NTPS =6
      GO TO 10
    6 NTPS =1
      GO TO 10
    7 IF(I-JMAX)8,8,5
    8 IF(KK-2)20,21,21
   20 K=KFO(KKK)
      GO TO 22
   21 K=KFO(KKK)
   22 IF(K-1)5,5,3
    3 IF(K-4)9,9,5
    9 NTPS =K
   10 IF(KK-2)35,36,36
   35 MJ(I)=IDKFO(NTPS)
      IF(NTPS-4)72,72,71
   71 NFLAG=NFLAG+1
   72 WRITE(MX,700)I,IDKFO(NTPS)
  700 FORMAT(8HOSECTION,I5,2X,2HIS,2X,A8/)
   11 CONTINUE
      WRITE(5'1)MJ
      RETURN
```

```
 36 NTY=NTPS                                                          ATP1 062
    LI(LL)=IDKFO(NTY)                                                 ATP1 063
    IDT=IPASS                                                         ATP1 064
    NDA(LL)=KNALB                                                     ATP1 065
    ILB=IDJB(LL,JB)                                                   ATP1 066
    IF(NTY-ILB) 85,40,85                                              ATP1 067
 40 IF(LL-3)86,41,41                                                  ATP1 068
 41 IF(ISV(LL))42,42,86                                               ATP1 069
 42 LI(LL)=KNALB                                                      ATP1 070
    GO TO 86                                                          ATP1 071
 85 NFLAG = NFLAG+1                                                   ATP1 072
    NDA(LL)=IDKFO(ILB)                                                ATP1 073
    IZ=IZ+1                                                           ATP1 074
 86 CONTINUE                                                          ATP1 075
    IF(IZ) 88,88,87                                                   ATP1 076
 87 IDT = IFAIL                                                       ATP1 077
 88 WRITE(MX,703)J,JB,MJ(J),(ISV(LL),LI(LL),NDA(LL),LL=1,3),IDT       ATP1 078
703 FORMAT(9HOEQUATION,I5,2X7HOF TYPE,2XI2,2XA4,                      ATP1 079
   13(3X,1H(,I2,1H,,2A4,1H),4X) ,  1X,A4)                             ATP1 080
    RETURN                                                            ATP1 081
    END                                                               ATP1 082
```

```
      SUBROUTINE BCGH(IS2,JMAX,KKK,ICON)
      DIMENSION SOL(41),BG(41),CE(41),KFO(1),KK1(4),AK1(31),AK3(5)
      MY=2
      MX=5
      READ(7'KKK)SOL
   50 FORMAT(4X,9E13.5)
      IS1 = JMAX - 2 * IS2
      DO1N=1,IS2
      N2=N*2
      CE(N)=SOL(N2)
      M=N2-1
      BG(N)=SOL(M)
C     WRITE(MX,30)N,N2,M,BG(N ),CE(N )
    1 CONTINUE
      DO2I=1,IS1
      K=I+2*IS2
      N=I+IS2
C  30 FORMAT (5HOBCGE , 3I5,2E20.5)
      READ(9'N)K1,G2,KFO
      KF=KFO(1)
      GO TO(11,12,13,12),KF
   11 WRITE(MX,20)
      RETURN
   12 BG(N)=SOL(K)
      CE(N)=0.0
C     WRITE(MX,30)I,K,N,BG(N),CE(N)
   20 FORMAT(14H ERROR IN BCGH )
      GO TO 2
   13 CE(N)=SOL(K)
      BG(N)=0.0
C     PRINT 30,I,K,N,BG(N),CE(N)
    2 CONTINUE
      DO 3 I=1,JMAX
      IF(KKK-1)4,4,5
    4 READ(9'I)I1,G1,I2,G2,G3,KK1,AK1
      GO TO 8
    5 READ(9'I)I1,G1,I2,G2,G3,KK1,AK1,A1,A2
C   8 WRITE(MX,30)I,BG(I),CE(I)
C  30 FORMAT(I4,2F15.5)
    8 IF(KKK-1)6,6,7
    6 WRITE(9'I)I1,G1,I2,G2,G3,KK1,AK1,BG(I),CE(I)
      GO TO 3
    7 WRITE(9'I)I1,G1,I2,G2,G3,KK1,AK1,A1,A2,BG(I),CE(I)
    3 CONTINUE
      IF(ICON-1)16,14,16
   14 DO 15 JJ=1,JMAX
      READ(9'JJ)K1,G1,KF,G2,G3,KK1,(AK1(I),I=1,23),DD,AK3,SD,VD,B,C,G,H
      READ(8'JJ)FAC,ZZ
      DD=0.
      SD=0.
      VD=0.
      G=0.
      H=0.
      FAC=0.
      ZZ=0.
      WRITE(9'JJ)K1,G1,KF,G2,G3,KK1,(AK1(I),I=1,23),DD,AK3,SD,VD,B,C,G,H
   15 WRITE(8'JJ)FAC,ZZ
   16 RETURN
      END
      SUBROUTINE BDOUT(IMAX,IDJOB,IDATE,ICON)                           BDOUT000
      REAL ID(41),IDJOB,IDATE                                           BDOUT001
      REAL NJD(1),NJU(1),JJS,JJE                                        BDOUT002
      DIMENSION PDEL(1),XJD(1),XJU(1),AK1(8),AK2(7),K2(4)               BDOUT003
      COMMON FAC1,FAC2,FAC3,ITYPE(35)                                   BDOUT004
      XMIN=1000.                                                        BDOUT005
      XMAX=0.                                                           BDOUT006
      NUM=0                                                             BDOUT007
      I=1                                                               BDOUT008
      MX=5                                                              BDOUT009
      READ(6'1)ID                                                       BDOUT010
      IF(ICON-4)2,2,1                                                   BDOUT011
    1 DO 29 JJ=1,IMAX                                                   BDOUT012
      READ(9'JJ)K1,NJU,KF,NJD,G1,K2,XJU,AK1,XJD,AK2,PDEL                BDOUT013
      JJS=NJU(1)                                                        BDOUT014
      JJE=NJD(1)                                                        BDOUT015
      WRITE(MX,21)                                                      BDOUT016
   21 FORMAT('1',40X,'COUPLED SYSTEM OF PREVIOUS 2 COMPONENTS')         BDOUT017
      WRITE(MX,32)IDJOB,IDATE                                           BDOUT018
   32 FORMAT(10X,A8,21X8HCOMPUTED 5XA8//1H0,2X,'SECT.',14X,             BDOUT019
     1'SECT.',15X,5HDIST.,15X 3HBOD,13X4HD.O ,'DEFICIT',12X,'JUNCTION'/ BDOUT020
     24X,'NO.',15X,'NAME',16X,'(MI)',14X,'(MG/L)',14X,'(MG/L)',17X,'ID'/BDOUT021
     33X,'-----',14X,'-----',14X,'-----',13X,'------',13X,11(1H-),12X,  BDOUT022
     48(1H-))                                                           BDOUT023
   22 NUM=NUM+1                                                         BDOUT024
      READ(4'NUM)G1,BBX,G1,DDX,G2,G3                                    BDOUT025
      BBX=BBX+G2                                                        BDOUT026
      DDX=DDX+G3                                                        BDOUT027
      WRITE(4'NUM)G1,BBX,G1,DDX                                         BDOUT028
      IF(G1-XJU(1))23,25,23                                             BDOUT029
   23 IF(G1-XJD(1))24,27,24                                             BDOUT030
   24 WRITE(MX,20)JJ,ID(JJ),G1,BBX,DDX                                  BDOUT031
      GO TO 22                                                          BDOUT032
   25 WRITE(MX,20)JJ,ID(JJ),G1,BBX,DDX,JJS                              BDOUT033
      IF(XJU(1)-XMIN)26,22,22                                           BDOUT034
   26 XMIN=XJU(1)                                                       BDOUT035
      GO TO 22                                                          BDOUT036
   27 WRITE(MX,20)JJ,ID(JJ),G1,BBX,DDX,JJE                              BDOUT037
      IF(XJD(1)-XMAX)28,29,29                                           BDOUT038
   28 XMAX=XJD(1)                                                       BDOUT039
   29 CONTINUE                                                          BDOUT040
      GO TO 31                                                          BDOUT041
    2 DO 3 JJ=1,IMAX                                                    BDOUT042
      READ(9'JJ)K1,NJU,KF,NJD,G1,K2,XJU,AK1,XJD,AK2,PDEL                BDOUT043
      JJS=NJU(1)                                                        BDOUT044
      JJE=NJD(1)                                                        BDOUT045
      WRITE(MX,30)ITYPE,IDJOB,IDATE                                     BDOUT046
      DELTA=1.0                                                         BDOUT047
      IF(PDEL(I))5,5,4                                                  BDOUT048
    4 DELTA=PDEL(I)                                                     BDOUT049
    5 IF(XJU(I)-XMIN)6,6,7                                              BDOUT050
    6 XMIN=XJU(I)                                                       BDOUT051
    7 CALL BOD3(JJ,XJU(1),BBX,DDX,KF)                                   BDOUT052
      WRITE(MX,20)JJ,ID(JJ),XJU(1),BBX,DDX,JJS                          BDOUT053
   20 FORMAT(1H0,I5,16X,A4,F20.1,2F20.4,17X,A3)                         BDOUT054
      NUM=NUM+1                                                         BDOUT055
      IF(ICON-4)34,33,34                                                BDOUT056
   33 READ(4'NUM)XJU,G1,XJU,G2                                          BDOUT057
   34 WRITE(4'NUM)XJU,BBX,XJU,DDX,G1,G2                                 BDOUT058
    8 XJU(1)=XJU(1)+DELTA                                               BDOUT059
      NUM=NUM+1                                                         BDOUT060
      IF(XJU(1)-XJD(1))9,10,10                                          BDOUT061
    9 CALL BODO(JJ,XJU(1),BBX,DDX,KF)                                BDOUT062
      WRITE(MX,20)JJ,ID(JJ),XJU(1),BBX,DDX                           BDOUT063
      IF(ICON-4)36,35,36                                             BDOUT064
   35 READ(4'NUM)XJU,G1,XJU,G2                                       BDOUT065
   36 WRITE(4'NUM)XJU,BBX,XJU,DDX,G1,G2                              BDOUT066
      GO TO 8                                                        BDOUT067
   10 IF(XJD(1)-XMAX)12,12,11                                        BDOUT068
   11 XMAX=XJD(1)                                                    BDOUT069
   12 CALL BODO(JJ,XJD(1),BBX,DDX,KF)                                BDOUT070
      WRITE(MX,20)JJ,ID(JJ),XJD(1),BBX,DDX,JJE                       BDOUT071
      IF(ICON-4)38,37,38                                             BDOUT072
   37 READ(4'NUM)XJU,G1,XJU,G2                                       BDOUT073
   38 WRITE(4'NUM)XJU,BBX,XJU,DDX,G1,G2                              BDOUT074
    3 CONTINUE                                                       BDOUT075
   30 FORMAT(1H1,43X,35A1/10X,A8,21X8HCOMPUTED 5XA8//1H0,2X,'SECT.',14X,BDOUT076
     1'SECT.',15X,5HDIST.,15X 3HBOD,13X4HD.O ,'DEFICIT',12X,'JUNCTION'/ BDOUT077
     24X,'NO.',15X,'NAME',16X,'(MI)',14X,'(MG/L)',14X,'(MG/L)',17X,'ID'/BDOUT078
     33X,'-----',14X,'-----',14X,'-----',13X,'-------',13X,11(1H-),12X,  BDOUT079
     48(1H-))                                                        BDOUT080
   31 RETURN                                                         BDOUT081
      END                                                            BDOUT082
      SUBROUTINE BODO(I,X,BBB,DDD,KF)
      DIMENSION FAC(1),ZZ(1)
      K=1
      READ(8'K)FAC,ZZ
      BBM=PFUN(K,X,1,1.,1.)
      YDIX=ZZ(1)+FAC(1)*BBM
      BBB=BBM+YB(K,X)
      DDM=PFUN(K,X,1,2.,1.)
      IF(KF-4)1,2,2
    1 DDD=DDM+YDIX
      GO TO 3
    2 DDD=DDM
    3 CONTINUE
C     PRINT 10,I,X,BBM,YDIX,DDM,BBB,DDD
   10 FORMAT (5H0BODO,I9, 6F16.4      )
      RETURN
      END
      SUBROUTINE BRIGH(LL,IS2,JMAX)
      DIMENSION FOR(41),AA(18),JBV(1),ISA(1),ISB(1),ISC(1),XSA(1),
     1XSB(1),XSC(1),W(1),Q(3),K1(4)
      MY=2
      MX=5
      DO 99 J=1,JMAX
      READ(9'J)L1,G1,K3,G2,G3,JBV,ISA,ISB,ISC,AA,XSA,XSB,XSC,W,QD,G1,
     1BDD
      IF(J-1)15,15,16
   15 READ(10'1)FOR
   16 JB=JBV(LL)
      I =ISA(LL)
      X=XSA(LL)
      L=NOSEC(I,IS2)
      K =ISB(LL)
      Y=XSB(LL)
      M=NOSEC(K,IS2)
      KK =ISC(LL)
      Z=XSC(LL)
      N=NOSEC(KK,IS2)
      READ(9'I)I1,G1,I2,G2,G3,K1,AU,A2,A3,A4,A5,Q(1)
      READ(9'K)I1,G1,I2,G2,G3,K1,AU,A2,A3,A4,A5,Q(2)
      IF(KK)18,18,17
   17 READ(9'KK)I1,G1,I2,G2,G3,K1,AU,A2,A3,A4,A5,Q(3)
C     CALL OUTLIST(BRIG)
   18 GO TO (1,1,1,4,5,5,5,8,9,10,11,11,13,13),JB
    1 FOR(J)=YB(K,Y)- YB(I,X)
      GO TO 99
    4 FOR(J) = -YB(I,X)
      GO TO 99
    5 FOR(J) = W(1)+Q(1)*YB(I,X)-Q(2)*YB(K, Y )
      GO TO 99
    8 FOR(J)=W(1)+Q(1)*YB(I,X)-Q(2)*YB(K,Y)-Q(3)*YB(KK,Z)
      GO TO 99
    9 FOR(J)=W(1)+Q(1)*YB(I,X)+Q(3)*YB(KK,Z)-Q(2)*YB(K,Y)
      GO TO 99
   10 FOR(J)=QD*BDD-Q(2)*YB(K,Y)
      GO TO 99
   11 FOR(J) =W (1)-Q(1)*YB(I,X)
      IF(JB-11)99,99,12
   12 FOR(J)=FOR(J)  -Q(2 )*YB(K ,Y)
      GO TO 99
   13 FOR(J)=W(1 )+Q(1)*YB(I,X)
      IF(JB-13)99,99,14
   14 FOR(J)=FOR(J)+Q(2)*YB(K,Y)
  200 FORMAT(/HOBRIGHT,2I5,E12.2)
C     PRINT 200,J,JB,FOR(J)
C  99 WRITE(MX,200)J,JB,FOR(J)
   99 CONTINUE
      WRITE(7'1)FOR
      RETURN
      END
      SUBROUTINE DATA1(NFLAG,NN,IDJOB,IDATE,IMAX,JMAX,IS2,K,IND,ICON)   DATA1000
      REAL MI,MJ(41),IDHJB(14),NJU(1),NJD(1),JNO(1)                     DATA1001
      REAL IDJOB,IDATE,ID(41),IDJ                                       DATA1002
      DIMENSION KFN(1),KFD(1),JBV(1),ISA(1),ISB(1),                     DATA1003
     1ISC(1),XJU(1),AREA(1),RK(1),AK(1),FF(1),Q(1),BD(1),WU(1),TIDE1(1),DATA1004
     2XJD(1),AREAD(1),DK(1),E(1),TEMP(1),PR(1),HT(1),VOL(1),PDEL(1),    DATA1005
     3XSA(1),XSB(1),XSC(1),W(1),QD(1),DD(1),BDD(1),BL(1),BC(1)          DATA1006
      COMMON FAC1,FAC2,FAC3,ITYPE(35)                                   DATA1007
      DATA IDHJB/'C 2D','C DN','C DP','C DC','M 2D','M DN','M DP',      DATA1008
     1'M3D1','M3D2','M DM','MCM1','MCM2','M1CM','M2CM'/                 DATA1009
      MY=2                                                              DATA1010
      MX=5                                                              DATA1011
      GO TO(33,34),K                                                    DATA1012
   33 IF(ICON)5,9,9                                                     DATA1013
    5 READ(MY,3)IDJOB,IDATE,IS2,IS1,NN,IND,ICON,FAC1,FAC2,FAC3,ITYPE    DATA1014
      GO TO 11                                                          DATA1015
    9 READ(MY,3)IDJOB,IDATE,IS2,IS1,NN,IND,NOTH,FAC1,FAC2,FAC3,ITYPE    DATA1016
      ICON=4                                                            DATA1017
    3 FORMAT(A4,1X,A4,1X,5I3,F5.0,2F6.0,3X,35A1)                        DATA1018
   11 WRITE(MX,157)                                                     DATA1019
  157 FORMAT('1JOB ID AND DATE CARD'/1X,20(1H-))                        DATA1020
      WRITE(MX,4)IDJOB,IDATE,IS2,IS1,NN,IND,ICON,FAC1,FAC2,FAC3,ITYPE   DATA1021
    4 FORMAT(1X,A4,1X,A4,5I5,3F8.3,1X,35A1)                             DATA1022
      IMAX=IS2+IS1                                                      DATA1023
      JMAX=2*IS2+IS1                                                    DATA1024
C     PRINT 90,IDJB                                                     DATA1025
      NFLAG=0                                                           DATA1026
      I=1                                                               DATA1027
      WRITE(MX,158)                                                     DATA1028
  158 FORMAT(' INPUT SECTION CARD LISTING'/1X,26(1H-))                  DATA1029
      DO 6 II=1,IMAX                                                    DATA1030
      READ(MY,10)NS2, J,KFN(I),NJU(I),XJU(I),AREA(I),RK(I),AK(I),FF(I), DATA1031
     1Q(I),BD(I),WU(I),TIDE1(I),IDJ                                     DATA1032
      WRITE(MX,20)NS2,J,KFN(I),NJU(I),XJU(I),AREA(I),RK(I),AK(I),FF(I), DATA1033
     1Q(I),BD(I),WU(I),TIDE1(I),IDJ                                     DATA1034
      IF(IDJ-IDJOB) 1,2,1                                               DATA1035
    1 WRITE(MX,30)J                                                     DATA1036
      NFLAG=NFLAG+1                                                     DATA1037
   30 FORMAT   (37H0ERROR IN IDENTIFICATION IN SECTION  I4//)           DATA1038
  330 FORMAT   (37H0ERROR IN IDENTIFICATION IN EQUATION I4//)           DATA1039
    2 READ(MY,10)NS1,J,KFD(I),NJD(I),XJD(I),AREAD(I),DK(I),E(I),TEMP(I),DATA1040
     1PR(I),HT(I),VOL(I),PDEL(I),ID(II)                                 DATA1041
      WRITE(MX,20)NS1,J,KFD(I),NJD(I),XJD(I),AREAD(I),DK(I),E(I),       DATA1042
     1TEMP(I),PR(I),HT(I),VOL(I),PDEL(I),ID(II)                         DATA1043
      WRITE(9'II)KFN,NJU,KFD,NJD,XJU,AREA,RK,AK,FF,Q,BD,WU,TIDE1,XJD,   DATA1044
     1AREAD,DK,E,TEMP,PR,HT,VOL,PDEL                                    DATA1045
      IF(AK(I)-RK(I))74,73,74                                           DATA1046
   73 NFLAG=NFLAG+1                                                     DATA1047
      WRITE(MX,701)J,AK(I),RK(I)                                        DATA1048
  701 FORMAT(24H0ERROR IN AK(I).OR.RK(I),2X I10,2E20.4)                 DATA1049
   74 IF(HT(I)) 6,75,6                                                  DATA1050
   75 NFLAG=NFLAG+1                                                     DATA1051
      WRITE(MX,702)J,HT(I)                                              DATA1052
  702 FORMAT(16H0ERROR IN HT(I) ,9X,I10,2E20.4)                         DATA1053
    6 CONTINUE                                                          DATA1054
   10 FORMAT(2X,2I3,I2, A3  ,F5.0,F7.0,3F5.0,4F8.0,4X,A4)               DATA1055
   20 FORMAT(1H0,3I4,3X,A3,F10.3,1X,F9.1,3F10.3,F9.1,F10.3,1X,E10.3,    DATA1056
     1F10.3,5X,A4/)                                                     DATA1057
      WRITE(6'1)ID                                                      DATA1058
      RETURN                                                            DATA1059
   34 J=1                                                               DATA1060
      WRITE(MX,159)                                                     DATA1061
```

```
159 FORMAT(' INPUT JUNCTION CARD LISTING'/1X,27(1H-))                 .DATA1062
    DO 8 JJ=1,JMAX                                                    DATA1063
    READ(9'JJ)KFN,NJU,KFD,NJD,XJU,AREA,RK,AK,FF,Q,BD,WU,TIDE1,XJD,    DATA1064
   1AREAD,DK,E,TEMP,PR,HT,VOL,PDEL                                    DATA1065
    READ(MY,50)I,JNO(J),JBV(J),ISA(J),XSA(J),ISB(J),XSB(J),ISC(J),    DATA1066
   1XSC(J),W(J),QD(J),DD(J),BDD(J),BL(J),BC(J),IDJ                    DATA1067
    WRITE(9'JJ)KFN,NJU,KFD,NJD,JNO,JBV,ISA,ISB,ISC,XJU,AREA,RK,AK,FF, DATA1068
   1Q,BD,WU,TIDE1,XJD,AREAD,DK,E,TEMP,PR,HT,VOL,PDEL,XSA,XSB,XSC,W,   DATA1069
   2QD,DD,BDD,BL,BC                                                   DATA1070
    JB=JBV(J)                                                         DATA1071
    MI=IDHJB(JB)                                                      DATA1072
    READ(5'2)MJ                                                       DATA1073
    MJ(JJ)=MI                                                         DATA1074
    WRITE(5'2)MJ                                                      DATA1075
    WRITE(MX,60)I,JNO(J),JBV(J),ISA(J),XSA(J),ISB(J),XSB(J),ISC(J),   DATA1076
   1XSC(J),W(J),QD(J),DD(J),BDD(J),BL(J),BC(J),IDJ                    DATA1077
 50 FORMAT(1XI2,A3,I3,3(I3,F6.1),6F6.0,4X,A4)                         DATA1078
 60 FORMAT(1H0,I4,1X,A3,I5, 3(1X,1H(,I5,F8.2,1H),1X),6F8.1,2X,A4/)    DATA1079
    IF(IDJOB-IDJ)    7,8,7                                            DATA1080
  7 WRITE(MX,330)JJ                                                   DATA1081
    NFLAG=NFLAG+1                                                     DATA1082
  8 CONTINUE                                                          DATA1083
    RETURN                                                            DATA1084
    END                                                               DATA1085
```

```
      SUBROUTINE DATA2(IMAX,JMAX,IS2,JBM)
      REAL NJU,NJD,JNO
      DIMENSION TEM(1),KTM(4),S(1),V(1),SD(1),VD(1),RK(1),AK(1),
     1DK(1),Q(1),AREAD(1),WU(1),HT(1),VOL(1),KFN(1),W(1),WD(1),
     2E(1),BD(1),AREA(1),TEMP(1),FF(1),PR(1)
      COMMON FAC1,FAC2,FAC3,ITYPE(35)
      MY=2
      MX=5
      I=1
      DO 55 KK=1,IMAX
      READ(9'KK)KFN,NJU,KFD,NJD,JNO,JBV,ISA,ISB,ISC,XJU,AREA,RK,AK,FF,
     1Q,BD,WU,TIDE1,XJD,AREAD,DK,E,TEMP,PR,HT,VOL,PDEL,XSA,XSB,XSC,W,
     2QD,DD,BDD,BL,BC
      AK(I)=AK(I)*FAC1**(TEMP(I)-20.)
      DK(I)=DK(I)*FAC2**(TEMP(I)-20.)
      RK(I)=RK(I)*FAC3**(TEMP(I)-20.)
      WRITE(9'KK)KFN,NJU,KFD,NJD,JNO,JBV,ISA,ISB,ISC,XJU,AREA,RK,AK,FF,
     1Q,BD,WU,TIDE1,XJD,AREAD,DK,E,TEMP,PR,HT,VOL,PDEL,XSA,XSB,XSC,W,
     2QD,DD,BDD,BL,BC
C     PRINT 2000,I,AK(I),DK(I),RK(I),TEMP(I),AREA(I),Q(I),
C    1WU(I),AREAD(I),HT(I),VOL(I),W(I),QD(I),DD(I),BC(I)
   55 CONTINUE
      CA=1.0/(5280.0**2)
      CV=1./(5280.**3)
      CF=3600.0*24.0/(5280.**3)
      CL=(1.0E+6)/(62.4*5280.**3)
      CM=12.0/39.37
      DO 5 KK=1,JMAX
      READ(9'KK)KFN,NJU,KFD,NJD,JNO,JBV,ISA,ISB,ISC,XJU,AREA,RK,AK,FF,
     1Q,BD,WU,TIDE1,XJD,AREAD,DK,E,TEMP,PR,HT,VOL,PDEL,XSA,XSB,XSC,W,
     2QD,DD,BDD,BL,BC
      IF(KK-IMAX)60,60,70
   60 AREA(I)=AREA(I)*CA
      Q(I)=Q(I)*CF
      WU(I)=WU(I)*CL
      AREAD(I)=AREAD(I)*CA
      HT(I)=HT(I)*CM
      VOL(I)=VOL(I)*CV
   70 W(I)=W(I)*CL
      QD(I)=QD(I)*CF
C     PRINT 2000,I,AK(I),DK(I),RK(I),TEMP(I),AREA(I),Q(I),
C    1WU(I),AREAD(I),HT(I),VOL(I),W(I),QD(I),DD(I),BC(I)
 2000 FORMAT(6HODATAC,I4, 3E14.2,11E9.2)
      WRITE(9'KK)KFN,NJU,KFD,NJD,JNO,JBV,ISA,ISB,ISC,XJU,AREA,RK,AK,FF,
     1Q,BD,WU,TIDE1,XJD,AREAD,DK,E,TEMP,PR,HT,VOL,PDEL,XSA,XSB,XSC,W,
     2QD,DD,BDD,BL,BC
    5 CONTINUE
      DO 32 KK=1,IMAX
      READ(9'KK)K1,G1,I1,G2,G3,KTM,A,B,RK,AK,FF,C,D,E,F,G,H,DK
      TEM(I)=DK(I)*FF(I)/(AK(I)-RK(I))
   32 WRITE(8'KK)TEM
      DO 33 KK=1,IMAX
      READ(8'KK)A1
      READ(9'KK)KFN,G1,I1,G2,G3,KTM,A,AREA,RK,AK,FF,B,BD,WU,C,D,E,DK,F,
     1G,PR,HT
      TEM(I)=0.
      IF(KFN(I) -1)33,31,33
   31 AA=BD(I)/HT(I)
      BB=DK(I)*WU(I)*FF(I)/(RK(I)*AREA (I))
      TEM(I)=(AA+PR(I)+BB)/AK(I)
   33 WRITE(8'KK)A1,TEM
      DO 1 KK=1,IMAX
      READ(8'KK)A1,A2                                                   DATA2062
      READ(9'KK)KFN,NJU,KFO,NJD,JNO,JBV,ISA,ISB,ISC,XJU,AREA,RK,AK,FF,  DATA2063
     1Q,BD,WU,TIDE1,XJD,AREAD,DK,E,TEMP,PR,HT,VOL,PDEL,XSA,XSB,XSC,W,   DATA2064
     2QD,DD,BDD,BL,BC                                                   DATA2065
      TEM(I)=Q(I)/AREA(I)                                               DATA2066
      CALL SVD(TEM(I),E(I),AK(I),SD(I),VD(I))                           DATA2067
      CALL SVD(TEM(I),E(I),RK(I),S(I),V(I))                             DATA2068
C     PRINT 10,I,AREA(I),U(I),E(I),AK(I),SD(I),VD(I)                    DATA2069
C     PRINT 10,I,Q(I),U(I),E(I),RK(I),S(I),V(I)                         DATA2070
      WRITE(9'KK)KFN,NJU,KFO,NJD,JNO,JBV,ISA,ISB,ISC,XJU,AREA,RK,AK,FF, DATA2071
     1Q,BD,WU,TIDE1,XJD,AREAD,DK,E,TEMP,PR,HT,VOL,PDEL,XSA,XSB,XSC,W,   DATA2072
     2QD,DD,BDD,BL,BC,S,V,SD,VD                                         DATA2073
      WRITE(8'KK)A1,A2,TEM                                              DATA2074
    1 CONTINUE                                                          DATA2075
   10 FORMAT(4H0SVD I5,7F17.6)                                          DATA2076
      WRITE(MX,20)(J,J=1,JMAX)                                          DATA2077
C  20 FORMAT(1H1 8X41I3)                                                DATA2078
   20 FORMAT('1',39X,'SYMBOLIC DO DEFICIT AND BOD MATRIX'/4X,'ROW',2X,  DATA2079
     1'JBV',24I4/13X,24I4)                                              DATA2080
      RETURN                                                            DATA2081
      END                                                               DATA2082
      SUBROUTINE DRIGH(LL,IS2,JMAX)                                     DRIGH000
      DIMENSION FOR(41),K1(4),AA(5),BB(14),QD(1),DD(1),FF(3),Q(3),      DRIGH001
     1DK(3),B(3),JBV(1),ISA(1),ISB(1),ISC(1),XSA(1),XSB(1),XSC(1),VOL(3)DRIGH002
      MY=2                                                              DRIGH003
      MX=5                                                              DRIGH004
      DO 99 J=1,JMAX                                                    DRIGH005
      READ(9'J)L1,G1,K3,G2,G3,JBV,ISA,ISB,ISC,BB,A1,A2,A3,A4,XSA,XSB,   DRIGH006
     1XSC,A5,QD,DD                                                      DRIGH007
      JB=JBV(LL)                                                        DRIGH008
      I =ISA(LL)                                                        DRIGH009
      X=XSA(LL)                                                         DRIGH010
      L=NOSEC(I,IS2)                                                    DRIGH011
      K =ISB(LL)                                                        DRIGH012
      Y=XSB(LL)                                                         DRIGH013
      M=NOSEC(K,IS2)                                                    DRIGH014
      KK =ISC(LL)                                                       DRIGH015
      Z=XSC(LL)                                                         DRIGH016
      N=NOSEC(KK,IS2)                                                   DRIGH017
      READ(9'I)I1,G1,I2,G2,G3,K1,A1,A2,A3,A4,FF(1),Q(1),AA,DK(1),A5,A6, DRIGH018
     1A7,A8,VOL(1),BB,B(1)                                              DRIGH019
      READ(9'K)I1,G1,I2,G2,G3,K1,A1,A2,A3,A4,FF(2),Q(2),AA,DK(2),A5,A6, DRIGH020
     1A7,A8,VOL(2),BB,B(2)                                              DRIGH021
      IF(KK)18,18,17                                                    DRIGH022
   17 READ(9'KK)I1,G1,I2,G2,G3,K1,A1,A2,A3,A4,FF(3),Q(3),AA,DK(3),A5,A6,DRIGH023
     1A7,A8,VOL(3),BB,B(3)                                              DRIGH024
   18 GO TO (1,1,1,4,5,5,5,5,5,10,11,12,13,14) , JB                     DRIGH025
    1 FOR(J)=YDP(K,Y,1) -YDP( I,X,1)                                    DRIGH026
      GO TO 99                                                          DRIGH027
    4 FOR(J)=-YDP(I,X,1)                                                DRIGH028
      GO TO 99                                                          DRIGH029
    5 FOR(J)=Q(1)*YDP(I,X,1)-           YDP(I,X,4)+YDP(K,Y,4)           DRIGH030
     1- Q(2)*YDP(K,Y,1)                                                 DRIGH031
      IF (JB-8)99,8,9                                                   DRIGH032
    8 FOR(J)=FOR(J)+YDP(KK,Z,4)-Q(3)*YDP(KK,Z,1)                        DRIGH033
      GO TO99                                                           DRIGH034
    9 FOR(J)=FOR(J)+Q(3)*YDP(KK,Z,1)-YDP(KK,Z,4)                        DRIGH035
      GO TO 99                                                          DRIGH036
   10 FOR(J)=              YDP(K,Y,4)-Q(2)*YDP(K,Y,1)+QD(1)*DD(1)       DRIGH037
      QD(1)=QD(1)                                                       DRIGH038
      DD(1)=DD(1)                                                       DRIGH039
      GO TO 99                                                          DRIGH040
   11 FOR(J)=YDP(I,X,4)-Q(1)*YDP(I,X,1)+FF(2)*DK(2)*B(2)*VOL(2)         DRIGH041
      GO TO 99                                                          DRIGH042
   12 FOR(J)=YDP(I,X,4)-Q(1)*YDP(I,X,1)+FF(3)*DK(3)*B(3)*VOL(3)         DRIGH043
      FOR(J)=FOR(J)+YDP(K,Y,4)-Q(1)*YDP(K,Y,1)                          DRIGH044
      GO TO 99                                                          DRIGH045
   13 FOR(J)=Q(1)*YDP(I,X,1)-YDP(I,X,4)+FF(2)*DK(2)*B(2)*VOL(2)         DRIGH046
      GO TO 99                                                          DRIGH047
   14 FOR(J)=Q(1)*YDP(I,X,1)-YDP(I,X,4)+FF(3)*DK(3)*B(3)*VOL(3)         DRIGH048
      FOR(J)=FOR(J)+Q(2)*YDP(K,Y,1)-YDP(K,Y,4)                          DRIGH049
  200 FORMAT(7H0DRIGHT,2I5,E12.2)                                       DRIGH050
C     PRINT 200,J,JB,FOR(J)                                             DRIGH051
C     WRITE(MX,200)J,JB,FOR(J)                                          DRIGH052
   99 CONTINUE                                                          DRIGH053
      WRITE(7'2)FOR                                                     DRIGH054
      RETURN                                                            DRIGH055
      END                                                               DRIGH056
      SUBROUTINE LST(LLL,KK,KKK,N)
      DIMENSION A(41),B(41)
      MX=5
      I=N+1
      IF(KKK-1)9,20,20
   20 IF(LLL-3)21,22,22
   21 MMM=1
      GO TO 23
   22 MMM=2
   23 IF(KKK-1)24,24,25
   24 GO TO(1,3,5,7),LLL
    1 WRITE(MX,2)
    2 FORMAT('1',51X,'INITIAL BOD MATRIX')
      GO TO 10
    3 WRITE(MX,4)
    4 FORMAT(////50X,'INVERTED BOD MATRIX')
      GO TO 10
    5 WRITE(MX,6)
    6 FORMAT('1',48X,'INITIAL DO DEFICIT MATRIX')
      GO TO 10
    7 WRITE(MX,8)
    8 FORMAT(////47X,'INVERTED DO DEFICIT MATRIX')
   10 WRITE(MX,1000)(J,J=1,N)
      DO 1001 I=1,N
      READ(MMM'I)(A(J),J=1,N)
 1001 WRITE(MX,2000)I,(A(J),J=1,N)
 1000 FORMAT(    /4X,9I13/)
   25 GO TO(11,14,11,14),LLL
   11 WRITE(MX,12)
   12 FORMAT(/' FORCING FUNCTIONS'/)
      GO TO 16
   14 WRITE(MX,15)
   15 FORMAT(/' SOLUTIONS FOR UNKNOWN COEFFICIENTS'/)
   16 READ(KK'MMM)(B(J),J=1,N)
      WRITE(MX,2000)I,(B(J),J=1,N)
 2000 FORMAT(I3,1X,9E13.5/(4X,9E13.5))
    9 RETURN
      END
      SUBROUTINE MATNV (LLL,N,M,KKK,KK)                                 MATNV000
      DIMENSION IPVOT(41),A(41),B(41),INDEX(41,2),PIVOT(41),C(10,41)    MATNV001
      DETRM=1.                                                          MATNV002
      IND=0                                                             MATNV003
      MX=3                                                              MATNV004
      IJK=1                                                             MATNV005
      LJK=LLL                                                           MATNV006
      DO 20 J=1,N                                                       MATNV007
   20 IPVOT(J)=0                                                        MATNV008
      READ(KK'KKK)(C(1,J),J=1,N)                                        MATNV009
   18 DO 550 I=1,N                                                      MATNV010
   10 WRITE(3,1)I                                                       MATNV011
    1 FORMAT(I5)                                                        MATNV012
      GO TO 380                                                         MATNV013
  110 IPVOT(ICOLM)=IPVOT(ICOLM)+1                                       MATNV014
                                                                        MATNV015
      INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL                 MATNV016
                                                                        MATNV017
  130 IF (IROW-ICOLM)140,260,140                                        MATNV018
  140 DETRM=DETRM*(-1.)                                                 MATNV019
      READ(LLL'IROW)A                                                   MATNV020
      READ(LLL'ICOLM)B                                                  MATNV021
  150 DO 200 L=1,N                                                      MATNV022
  160 SWAP=A(L)                                                         MATNV023
  170 A(L)=B(L)                                                         MATNV024
  200 B(L)=SWAP                                                         MATNV025
      WRITE(LLL'IROW)A                                                  MATNV026
      WRITE(LLL'ICOLM)B                                                 MATNV027
  205 IF(M) 260, 260, 210                                               MATNV028
  210 DO 250 L=1, M                                                     MATNV029
  220 SWAP=C(L,IROW)                                                    MATNV030
  230 C(L,IROW)=C(L,ICOLM)                                              MATNV031
  250 C(L,ICOLM)=SWAP                                                   MATNV032
  260 INDEX(I,1)=IROW                                                   MATNV033
  270 INDEX(I,2)=ICOLM                                                  MATNV034
  310 IF(I-N)111,550,550                                                MATNV035
  111 READ(LLL'ICOLM)A                                                  MATNV036
      PIVOT(I)=A(ICOLM)                                                 MATNV037
  320 DETRM=DETRM*PIVOT(I)                                              MATNV038
                                                                        MATNV039
      DIVIDE PIVOT ROW BY PIVOT ELEMENT                                 MATNV040
                                                                        MATNV041
  330 A(ICOLM)=1.                                                       MATNV042
  340 DO 350 L=1,N                                                      MATNV043
  350 A(L)=A(L)/PIVOT(I)                                                MATNV044
      WRITE(LLL'ICOLM)A                                                 MATNV045
  355 IF(M) 380, 380, 360                                               MATNV046
  360 DO 370 L=1,M                                                      MATNV047
  370 C(L,ICOLM)=C(L,ICOLM)/PIVOT(I)                                    MATNV048
      IND=IND+1                                                         MATNV049
      IF(IND-1)380,380,550                                              MATNV050
                                                                        MATNV051
      REDUCE NON-PIVOT ROWS                                             MATNV052
                                                                        MATNV053
                                                                        MATNV054
      SEARCH FOR PIVOT ELEMENT                                          MATNV055
                                                                        MATNV056
  380 AMAX=0.0                                                          MATNV057
      DO 105 J=1,N                                                      MATNV058
      READ(LLL'J)A                                                      MATNV059
      IF(IND-1)82,40,82                                                 MATNV060
   82 IF(I-1)31,31,40                                                   MATNV061
   40 IF(J-1)5,5,6                                              MATNV062
    5 READ(LLL'ICOLM)B                                          MATNV063
    6 IF(J-N)2,4,4                                              MATNV064
    2 JNU=J+1                                                   MATNV065
      FIND(LLL'JNU)                                             MATNV066
    4 IF(J-ICOLM)400,31,400                                     MATNV067
  400 T=A(ICOLM)                                                MATNV068
      A(ICOLM)=0.                                               MATNV069
  430 DO 450 L=1,N                                              MATNV070
  450 A(L)=A(L)-B(L)*T                                          MATNV071
      IF(M)31,31,460                                            MATNV072
  460 DO 500 L=1,M                                              MATNV073
  500 C(L,J)=C(L,J)-C(L,ICOLM)*T                                MATNV074
   31 IF(IJK-1)32,32,105                                        MATNV075
   32 IF(IPVOT(J)-1)60,105,60                                   MATNV076
   60 DO 100 K=1,N                                              MATNV077
      IF(IPVOT(K)-1)80,100,33                                   MATNV078
   33 IJK=2                                                     MATNV079
      GO TO 105                                                 MATNV080
   80 IF (ABS(AMAX)-ABS(A(K)))85,100,100                        MATNV081
   85 JROW=J                                                    MATNV082
      JCOLM=K                                                   MATNV083
   95 AMAX=A(K)                                                 MATNV084
  100 CONTINUE                                                  MATNV085
  105 WRITE(LLL'J)A                                             MATNV086
   24 IF(IJK-1)34,34,760                                        MATNV087
   34 IROW=JROW                                                 MATNV088
      ICOLM=JCOLM                                               MATNV089
      IF(IND-1)110,560,110                                      MATNV090
  560 IND=2                                                     MATNV091
      GO TO 110                                                 MATNV092
  550 CONTINUE                                                  MATNV093
:                                                               MATNV094
:     INTERCHANGE COLUMNS                                       MATNV095
:                                                               MATNV096
      WRITE(KK'KKK)(C(1,J),J=1,N)                               MATNV097
   19 M1=1                                                      MATNV098
      N1=10                                                     MATNV099
      JJ1=10                                                    MATNV100
      IF(N1-N)9,9,16                                            MATNV101
   16 N1=N                                                      MATNV102
      JJ1=N                                                     MATNV103
    9 I1=0                                                      MATNV104
      DO 3 K=M1,N1                                              MATNV105
      READ(LLL'K)A                                              MATNV106
      I1=I1+1                                                   MATNV107
      DO 3 J1=1,N                                               MATNV108
    3 C(I1,J1)=A(J1)                                            MATNV109
  600 DO 710 I=1,N                                              MATNV110
  610 L=N+1-I                                                   MATNV111
  620 IF (INDEX(L,1)-INDEX(L,2)) 630, 710, 630                  MATNV112
  630 JROW=INDEX(L,1)                                           MATNV113
  640 JCOLM=INDEX(L,2)                                          MATNV114
  650 DO 705 K=1,JJ1                                            MATNV115
  660 SWAP=C(K,JROW)                                            MATNV116
  670 C(K,JROW)=C(K,JCOLM)                                      MATNV117
  700 C(K,JCOLM)=SWAP                                           MATNV118
  705 CONTINUE                                                  MATNV119
  710 CONTINUE                                                  MATNV120
      LLL=LJK                                                   MATNV121
      I=0                                                       MATNV122
   11 DO 14 K=M1,N1                                             MATNV123
```

```
      I=I+1                                                             MATNV124
      DO 12 J1=1,N                                                      MATNV125
   12 A(J1)=C(I,J1)                                                     MATNV126
   14 WRITE(LLL'K)A                                                     MATNV127
      IF(N1-N)21,760,760                                                MATNV128
   21 M1=M1+10                                                          MATNV129
      N1=N1+10                                                          MATNV130
      IF(N1-N)9,9,22                                                    MATNV131
   22 JJ1=N-N1                                                          MATNV132
      N1=N                                                              MATNV133
      GO TO 9                                                           MATNV134
  760 RETURN                                                            MATNV135
      END                                                               MATNV136
      SUBROUTINE MATS(K1,K2,K3,K4,MM,IS2,JMAX,IND)
      DIMENSION TM(41),Q(3),VOL(3),TIDE1(3),ARK(3),JBV(1),ISA(1),
     1XSA(1),ISB(1),XSB(1),ISC(1),XSC(1),SD(1),VD(1),AA(7),BB(6),M1(4)
      DO 99 NN=1,JMAX
      LL=1
      READ(9'NN)L1,G1,L3,G2,G3,JBV,ISA,ISB,ISC,AA,BB,A1,A2,A3,A4,A5,
     1XSA,XSB,XSC
      MX=3
      DO 40 I=1,41
   40 TM(I)=0.
      JB=JBV(LL)
      I=ISA(LL)
      X=XSA(LL)
      L=NOSEC(I,IS2)
      IM=L-1
      K=ISB(LL)
      Y=XSB(LL)
      M=NOSEC(K,IS2)
      KM=M-1
      KK=ISC(LL)
      Z=XSC(LL)
      N=NOSEC(KK,IS2)
      KKM= N-1
      SD(1)=IND
      VD(1)=0.
      IF(IND-1)24,24,25
   24 READ(9'I)I1,G1,I2,G2,G3,M1,A1,A2,ARK(1),A3,A4,Q(1),A5,A6,TIDE1(1),
     1AA,VOL(1)
      READ(9'K)I1,G1,I2,G2,G3,M1,A1,A2,ARK(2),A3,A4,Q(2),A5,A6,TIDE1(2),
     1AA,VOL(2)
      IF(KK)26,26,27
   27 READ(9'KK)I1,G1,I2,G2,G3,M1,A1,A2,ARK(3),A3,A4,Q(3),A5,A6,
     1TIDE1(3),AA,VOL(3)
      GO TO 26
   25 READ(9'I)I1,G1,I2,G2,G3,M1,A1,A2,A3,ARK(1),A4,Q(1),A5,A6,TIDE1(1),
     1AA,VOL(1)
      READ(9'K)I1,G1,I2,G2,G3,M1,A1,A2,A3,ARK(2),A4,Q(2),A5,A6,TIDE1(2),
     1AA,VOL(2)
      IF(KK)26,26,21
   21 READ(9'KK)I1,G1,I2,G2,G3,M1,A1,A2,A3,ARK(3),A4,Q(3),A5,A6,
     1TIDE1(3),AA,VOL(3)
   26 GO TO(1,2,3,4,5,6,7,8,9,10,11,12,13,14),JB
    1 TM(IM)=PFUN(I,X,K1,SD,VD)
      TM(L)=PFUN(I,X,K2,SD,VD)
      TM(KM)=-PFUN(K,Y,K1,SD,VD)
      TM(M)=-PFUN(K,Y,K2,SD,VD)
      GO TO  99
    2 TM(L   )=PFUN(I,X,K1,SD,VD)
      TM(KM   )=-PFUN(K,Y,K1,SD,VD)
      TM(M   )=-PFUN(K,Y,K2,SD,VD)
      GO TO  99
    3 TM(IM)=PFUN(I,X,K1,SD,VD)
      TM(L)=PFUN(I,X,K2,SD,VD)
      TM(M)=-PFUN(K,Y,K2,SD,VD)
      GO TO  99
    4 TM(IM)=PFUN(I,X,K1,SD,VD)
      TM(L)=PFUN(I,X,K2,SD,VD)
      TM(M)=-1.0
      GO TO 99
    5 TM(IM)=PFUN(I,X,K3,SD,VD)-Q(1)*PFUN(I,X,K1,SD,VD)
      TM(L)=PFUN(I,X,K4,SD,VD)-Q(1)*PFUN(I,X,K2,SD,VD)
      TM(KM)=Q(2)*PFUN(K,Y,K1,SD,VD)-PFUN(K,Y,K3,SD,VD)
      TM(M )=Q(2)*PFUN(K,Y,K2,SD,VD)-PFUN(K,Y,K4,SD,VD)               MATS 062
      GO TO 99                                                        MATS 063
    6 TM(L)=PFUN(I,X,K3,SD,VD)-Q(1)*PFUN(I,X,K1,SD,VD)                MATS 064
      TM(KM   )=Q(2)*PFUN(K,Y,K1,SD,VD)-PFUN(K,Y,K3,SD,VD)            MATS 065
      TM(M   )=Q(2)*PFUN(K,Y,K2,SD,VD)-PFUN(K,Y,K4,SD,VD)             MATS 066
      GO TO 99                                                        MATS 067
    7 TM(IM)=PFUN(I,X,K3,SD,VD)-Q(1)*PFUN(I,X,K1,SD,VD)               MATS 068
      TM(L)=PFUN(I,X,K4,SD,VD)-Q(1)*PFUN(I,X,K2,SD,VD)                MATS 069
      TM(M     )=Q(2)*PFUN(K,Y,K2,SD,VD)-PFUN(K,Y,K4,SD,VD)           MATS 070
      GO TO 99                                                        MATS 071
    8 TM(IM   )=PFUN(I,X,K3,SD,VD)-Q(1)*PFUN(I,X,K1,SD,VD)            MATS 072
      TM(L   )=PFUN(I,X,K4,SD,VD)-Q(1)*PFUN(I,X,K2,SD,VD)             MATS 073
      TM(KM   )=Q(2)*PFUN(K,Y,K1,SD,VD)-PFUN(K,Y,K3,SD,VD)            MATS 074
      TM(M   )=Q(2)*PFUN(K,Y,K2,SD,VD)-PFUN(K,Y,K4,SD,VD)             MATS 075
   28 TM( N   ) = Q(3)*PFUN(KK,Z,K2,SD,VD)-PFUN(KK,Z,K4,SD,VD)        MATS 076
      TM(KKM   )=Q(3)*PFUN(KK,Z,K1,SD,VD)-PFUN(KK,Z,K3,SD,VD)         MATS 077
      GO TO 99                                                        MATS 078
    9 TM(IM   )=PFUN(I,X,K3,SD,VD)-Q(1)*PFUN(I,X,K1,SD,VD)            MATS 079
      TM(L   )=PFUN(I,X,K4,SD,VD)-Q(1)*PFUN(I,X,K2,SD,VD)             MATS 080
      TM(KM   )=Q(2)*PFUN(K,Y,K1,SD,VD)-PFUN(K,Y,K3,SD,VD)            MATS 081
      TM(M   )=Q(2)*PFUN(K,Y,K2,SD,VD)-PFUN(K,Y,K4,SD,VD)             MATS 082
   29 TM(KKM   )=PFUN(KK,Z,K3,SD,VD)-Q(3)*PFUN(KK,Z,K1,SD,VD)         MATS 083
      TM(N   )=PFUN(KK,Z,K4,SD,VD)-Q(3)*PFUN(KK,Z,K2,SD,VD)           MATS 084
      GO TO 99                                                        MATS 085
   10 TM(KM   )=Q(2)*PFUN(K,Y,K1,SD,VD)-PFUN(K,Y,K3,SD,VD)            MATS 086
      TM(M   )=Q(2)*PFUN(K,Y,K2,SD,VD)-PFUN(K,Y,K4,SD,VD)             MATS 087
C     PRINT 200,I,J,TM(L,J),X,K,J,TM(M,J),Y,KK,J,TM( N,J),Z           MATS 088
      GO TO 99                                                        MATS 089
   11 TM(IM   ) = Q(1)*PFUN(I,X,K1,SD,VD)-PFUN(I,X,K3,SD,VD)          MATS 090
      TM(L   )=Q(1)*PFUN(I,X,K2,SD,VD)-PFUN(I,X,K4,SD,VD)             MATS 091
      JBB=JB-10                                                       MATS 092
      TM(M   )=VOL(2)*ARK(2)                                          MATS 093
      GO TO 99                                                        MATS 094
   12 TM(IM   )=Q(1)*PFUN(I,X,K1,SD,VD)-PFUN(I,X,K3,SD,VD)            MATS 095
      TM(L   )=Q(1)*PFUN(I,X,K2,SD,VD)-PFUN(I,X,K4,SD,VD)             MATS 096
      TM(KM   )=Q(2)*PFUN(K,Y,K1,SD,VD)-PFUN(K,Y,K3,SD,VD)            MATS 097
      TM(M   )=Q(2)*PFUN(K,Y,K2,SD,VD)-PFUN(K,Y,K4,SD,VD)             MATS 098
      TM(N)=VOL(3)*ARK(3)                                             MATS 099
      GO TO 99                                                        MATS 100
   13 TM(IM   )=PFUN(I,X,K3,SD,VD)-Q(1)*PFUN(I,X,K1,SD,VD)            MATS 101
      TM(L   )=PFUN(I,X,K4,SD,VD)-Q(1)*PFUN(I,X,K2,SD,VD)             MATS 102
      JBB=JB-12                                                       MATS 103
      TM(M   )=Q(2)+VOL(2)*(TIDE1(2)+ARK(2))                          MATS 104
      GO TO 99                                                        MATS 105
   14 TM(IM   ) =PFUN(I,X,K3,SD,VD)-Q(1)*PFUN(I,X,K1,SD,VD)           MATS 106
      TM(L   )=PFUN(I,X,K4,SD,VD)-Q(1)*PFUN(I,X,K2,SD,VD)             MATS 107
      TM(KM   )=PFUN(K,Y,K3,SD,VD)-Q(2)*PFUN(K,Y,K1,SD,VD)            MATS 108
      TM(M   )=PFUN(K,Y,K4,SD,VD)-Q(2)*PFUN(K,Y,K2,SD,VD)             MATS 109
      TM(N)=Q(3)+VOL(3)*(TIDE1(3)+ARK(3))                             MATS 110
   99 WRITE(MM'NN)TM                                                  MATS 111
      RETURN                                                          MATS 112
      END                                                             MATS 113
      SUBROUTINE MEMS(JMAX,KK,LL)                                       MEMS 000
      DIMENSION A(41),B(41),C(41)                                       MEMS 001
      J=1                                                               MEMS 002
      READ(7'KK)B                                                       MEMS 003
      DO 2 I=1,JMAX                                                     MEMS 004
      READ(LL'I)A                                                       MEMS 005
      S=0.                                                              MEMS 006
      DO 1 K=1,JMAX                                                     MEMS 007
    1 S=S+A(K)*B(K)                                                     MEMS 008
    2 C(I)=S                                                            MEMS 009
      WRITE(7'KK)C                                                      MEMS 010
      RETURN                                                            MEMS 011
      END                                                               MEMS 012
```

```
                                                                          NOSEC000
      FUNCTION NOSEC(IL,IS2)                                              NOSEC001
      IF(IL-IS2)1,1,2                                                     NOSEC002
    1 NOSEC=2*IL                                                          NOSEC003
      RETURN                                                              NOSEC004
    2 NOSEC=2*IS2+(IL-IS2)                                                NOSEC005
      RETURN                                                              NOSEC006
      END
      FUNCTION PFUN(J,X,K,ABC,DEF)
      DIMENSION SSD(1),VVD(1),KFN(1),WU(1),AREA(1),RK(1),E(1),XJD(1),
     1AREAD(1),K1(4),BB(14),KFO(1),B(1),C(1)
      I=1
      IF(ABC-1.)17,17,18
   17 READ(9'J)KFN,G1,KFO,G2,G3,K1,A1,AREA,RK,A2,A3,A4,A5,WU,A6,XJD,
     1AREAD,A7,E,BB,SSD,VVD,A8,A9,B,C
      GO TO 19
   18 READ(9'J)KFN,G1,KFO,G2,G3,K1,A1,AREA,RK,A2,A3,A4,A5,WU,A6,XJD,
     1AREAD,A7,E,BB,A8,A9,SSD,VVD,A10,A11,B,C
   19 IF(DEF)2,2,3
    2 L=K
      IF(K-5)9,9,1
    1 L=K-4
    9 KF=KFN(I)
      GO TO (10,20,30),KF
   10 Y=1.
      GO TO (11,12,13,14,15,16) ,L
   11 PFUN = EXP (X*SSD(I))/Y
C     PRINT 200,K,KF,I,X,SSD(I),PFUN
  200 FORMAT(I5,4F12.2)
      GO TO 4
   12 PFUN = EXP (X*VVD(I))/Y
C     PRINT 200,K,KF,I,X,VVD(I),PFUN
      GO TO 4
   13 PFUN = WU(I)/(AREA(I)*RK(I))
      GO TO 4
   14 PFUN=SSD(I)*EXP (X*SSD(I))*E(I)*AREA(I)
      GO TO 4
   15 PFUN = VVD(I)*EXP (X*VVD(I))*E(I)*AREA(I)
      GO TO 4
   16 PFUN = 0.0
      GO TO 4
   20 AA=AREA(I)
      Y=X
      IF(X-XJD(I)) 40,40,39
   40 AA=AREAD(I)
   39 GO TO (11,12,16,24,25,16),L
   24 PFUN=(SSD(I)-1.0/X)*EXP (SSD(I)*X)/X
      PFUN=PFUN*AA *E(I)
C     PRINT 200,K,KF,I,X,SSD(I),PFUN
      GO TO 4
   25 PFUN=(VVD(I)-1.0/X)*EXP (VVD(I)*X)/X
      PFUN=PFUN*AA*E(I)
C     PRINT 200,K,KF,I,X,VVD(I),PFUN
C     RETURN
      GO TO 4
    3 M=KFO(1)
      IF(M-3)26,27,26
   26 L=K
      GO TO 9
   27 L=K+1
      GO TO 9
    4 IF(DEF)30,30,5
    5 GO TO(21,8,7,6),M
    6 PFUN=B(1)
      GO TO 30
    7 PFUN=PFUN*C(1)
      GO TO 30
    8 PFUN=PFUN*B(1)
      GO TO 30
   21 IF(L-K)22,22,23
   22 L=K+1
      ASUM=PFUN*B(1)                                          PFUN 062
      GO TO 9                                                 PFUN 063
   23 PFUN=PFUN*C(1)+ASUM                                     PFUN 064
   30 RETURN                                                  PFUN 065
      END                                                     PFUN 066
                                                              PFUN 067
```

```
                                                                 PLAC1000
      SUBROUTINE PLAC1(IS2,JMAX,JBM)                             PLAC1001
      REAL JNO(1),LINE(41),J                                     PLAC1002
      DIMENSION ISA(1),ISB(1),ISC(1),JBV(1)                      PLAC1003
      DATA LITRL/' -'/                                           PLAC1004
      MX=5                                                       PLAC1005
      IS2M=IS2*2                                                 PLAC1006
      IS2 = IS2M/2                                               PLAC1007
      JJ=1                                                       PLAC1008
      DO 6 KK=1,JMAX                                             PLAC1009
      READ(9'KK)K1,G1,K3,G2,JNO,JBV,ISA,ISB,ISC                  PLAC1010
      J=JNO(JJ)                                                  PLAC1011
      DO 7 L=1,JBM                                               PLAC1012
    7 LINE(L)=LITRL                                              PLAC1013
      DO 4 LL=1,3                                                PLAC1014
      GO TO(10,11,12),LL                                         PLAC1015
   10 KS=ISA(JJ)                                                 PLAC1016
      GO TO 14                                                   PLAC1017
   11 KS=ISB(JJ)                                                 PLAC1018
      GO TO 14                                                   PLAC1019
   12 KS=ISC(JJ)                                                 PLAC1020
   14 IF(KS) 4,4,1                                               PLAC1021
    1 IS=2*KS                                                    PLAC1022
      IF(IS-IS2M) 2,2,3                                          PLAC1023
    2 LINE(IS)=J                                                 PLAC1024
      LINE(IS-1)=J                                               PLAC1025
      GO TO 4                                                    PLAC1026
    3 IS=IS2+KS                                                  PLAC1027
      LINE(IS)=J                                                 PLAC1028
    4 CONTINUE                                                   PLAC1029
      WRITE(MX,30)KK,JBV(JJ), (LINF(I),I=1,JMAX)                 PLAC1030
   30 FORMAT(1H0,2I5,2X,24(1XA3)/13X,24(1X,A3))                  PLAC1031
C  30 FORMAT(1H0,2I4,40A3)                                       PLAC1032
    6 CONTINUE                                                   PLAC1033
      RETURN                                                     PLAC1034
      END
      SUBROUTINE PR1(K,IMAX,JMAX,IDJOB,IDATE,NK)
      REAL MI(41),ID(41),NJU(1),NJD(1),IDJOB,IDATE,JNO(1)
      DIMENSION KFN(1),KFO(1),JBV(1),ISA(1),ISB(1),
     1ISC(1),XJU(1),AREA(1),RK(1),AK(1),FF(1),Q(1),BD(1),WU(1),TIDE1(1),
     2XJD(1),AREAD(1),DK(1),E(1),TEMP(1),PR(1),HT(1),VOL(1),PDEL(1),
     3XSA(1),XSB(1),XSC(1),W(1),QD(1),DD(1),BDD(1),BL(1),BC(1)
      DIMENSION S(1),V(1),SD(1),VD(1),B(1),C(1),G(1),H(1)
      COMMON FAC1,FAC2,FAC3,ITYPE(35)
      READ(6'1)ID
      MY=2
      MX=5
      GO TO(5,6,7),K
    5 IF(NK)8,8,9
    8 WRITE(MX,14)ITYPE,IDJOB,IDATE
   14 FORMAT('1',42X,35A1/' MODEL RUN ',A8,'  COMPUTED',2X,A8,2X,
     130HDATA GIVEN IN  ORIGINAL  UNITS/)
      GO TO 11
    9 WRITE(MX,15)ITYPE,IDJOB,IDATE
   15 FORMAT('1',42X,35A1/' MODEL RUN ',A8,2X,'COMPUTED',2X,A8,2X
     1'DATA GIVEN IN COMPATIBLE UNITS'/)
   11 WRITE(MX,10)
      READ(5'1)MI
      I=1
      DO 1 J=1,IMAX
      READ(9'J)KFN,NJU,KFO,NJD,JNO,JBV,ISA,ISB,ISC,XJU,AREA,RK,AK,FF,
     1Q,BD,WU,TIDE1,XJD,AREAD,DK,E,TEMP,PR,HT,VOL,PDEL,XSA,XSB,XSC,W,
     2QD,DD,BDD,BL,BC
   10 FORMAT(///5X7HSECTION,4H KFN,4H KFO,2X,'JUNCTION ID AND DISTANCE',
     112X,2HRK,8X2HAK,8X2HDK,9X1HE,8X2HFF,6X,4HTEMP3X,'PDEL'/28X,'AT',
     2' ENDS',20X,'REM.RATE ',' REA.RATE ',' DEO.RATE ',' DISP.RATE ',
     3' ULT/5DAY ',4X,'  MILE-INCR'/56X,'(1/DAY)',3X,'(1/DAY)',3X,
     4'(1/DAY)',2X,'(MI2/DAY)',11X,'(CENT)',2X,'(MI)'/)
    1 WRITE(MX,20)J,ID(J),MI(J),KFN(I),KFO(I),NJD(I),XJD(I),NJU(I),
     1XJU(I),RK(I),AK(I),DK(I),E(I), FF(I),TEMP(I),PDEL(I)
   20 FORMAT(1H0,I2,1XA4,1XA4,2I4,2(1X,1H(,A3,1H,F6.1,1H),2X),6F10.3,
     1F7.3)
      WRITE(MX,30)
   30 FORMAT(///5X7HSECTION,6X4HAREA,7X5HAREAD,9X1HQ,11X2HBD,10X2HPR,
     18X5HTIDE1,9X,2HHT,7X,3HVOL,8X,'WU/AREA*RK'/17X,'UPSTRM',5X,
     2'DWNSTRM',7X,'FLOW',5X,'BENTHL-DMD',3X,'PHOTO-RESP',2X,'TIDE-COEF'
     35X,'DEPTH',4X,'VOLUME',8X,'UNF-LOAD')
      IF(NK)21,21,23
   21 WRITE(MX,22)
   22 FORMAT(17X,'(FT**2)',4X,'(FT**2)',7X,'(CFS)',3X,'(GMS/M2-DAY)',
     12X,'(MG/L-DAY)',16X,'(FT.)',3X,'(FT**3)',4X,'(LBS/DAY-MI)')
      GO TO 16
   23 WRITE(MX,24)
   24 FORMAT(17X,'(MI**2)',4X,'(MI**2)'4X,'(MI**3/DAY)',2X,
     1'(GMS/M2-DAY)',2X,'(MG/L-DAY)',15X,'(MTRS)',2X,'(MI**3)',8X,
     2'(MG/L)')
   16 DO 2 J=1,IMAX
      READ(9'J)KFN,NJU,KFO,NJD,JNO,JBV,ISA,ISB,ISC,XJU,AREA,RK,AK,FF,
     1Q,BD,WU,TIDE1,XJD,AREAD,DK,E,TEMP,PR,HT,VOL,PDEL,XSA,XSB,XSC,W,
     2QD,DD,BDD,BL,BC
    2 WRITE(MX,40)J, ID(J), MI(J),AREA(I),AREAD(I),Q(I),BD(I),
     1PR(I),TIDE1(I),HT(I),VOL(I),WU(I)
      RETURN
   40 FORMAT(1H0,I2,1XA4,1XA4,5E12.3,3E11.3,2X,E12.3)
    6 WRITE(MX,50)
      READ(5'1)MI
      DO 3 I=1,IMAX
      READ(9'I)KFN,NJU,KFO,NJD,JNO,JBV,ISA,ISB,ISC,XJU,AREA,RK,AK,FF,
      1Q,BD,WU,TIDE1,XJD,AREAD,DK,E,TEMP,PR,HT,VOL,PDEL,XSA,XSB,XSC,W,
      2QD,DD,BDD,BL,BC,S,V,SD,VD,B,C,G,H
       READ(8'I)FAC,ZZ,U
   50 FORMAT(///5X7HSECTION,8X,1HU,9X1HS,9X1HV,8X2HSD,
      18X2HVD,9X1HB,9X1HC,9X1HG,9X1HH,5X3HFAC6X2HZZ/)
    3 WRITE(MX,60)I,ID(I),MI(I),U,S(1),V(1),SD(1),VD(1),B(1),C(1) ,
      1G(1),H(1),FAC,ZZ
   60 FORMAT(1H0,I2,1XA4,1XA4,9F10.5,2F8.2)
       RETURN
    7 WRITE(MX,70)
   70 FORMAT(/1H05X8HEQUATION,4X,'JUNCTION ID, SECTIONS',4X,
      117X1HW,13X2HQD,10X2HDD,9X3HBDD/18X,'MEETING AT ',
      2'JUNCTION,',19X,'PT-LOAD',7X,'FLOW-DAM',5X,'DEF-DAM',4X,'BOD-DAM')
       IF(NK)26,26,28
   26 WRITE(MX,27)
   27 FORMAT(18X,'AND DISTANCE',26X,'(LBS/DAY)',7X,'(CFS)',7X,'(MG/L)',
      15X,'(MG/L)')
       GO TO 17
   28 WRITE(MX,29)
   29 FORMAT(18X,'AND DISTANCE',21X,'(MI**3/DAY)(MG/L)',2X,'(MI**3/DAY)'
      14X,'(MG/L)',5X,'(MG/L)')
   17 READ(5'2)MI
       J=1
       DO 4 I=1,JMAX
       READ(9'I)KFN,NJU,KFO,NJD,JNO,JBV,ISA,ISB,ISC,XJU,AREA,RK,AK,FF,
      1Q,BD,WU,TIDE1,XJD,AREAD,DK,E,TEMP,PR,HT,VOL,PDEL,XSA,XSB,XSC,W,
      2QD,DD,BDD,BL,BC
    4  WRITE(MX,80)I,JBV(J),MI(I),JNO(J),ISA(J),XSA(J),ISB(J),
      1XSB(J),ISC(J),XSC(J),W(J),QD(J),DD(J),BDD(J)
   80 FORMAT(1H0,I2,I3,1XA4,1XA3,1X,3(1H(,I3,F5.1,1H)),6XF12.5,3X,
      13F12.5)
       RETURN
       END
      SUBROUTINE SVD(U,E,A,S,V)                                         SVD  000
      IF(U) 2,1,2                                                       SVD  001
    1 S=SQRT (A/E)                                                      SVD  002
      V=-S                                                              SVD  003
      RETURN                                                            SVD  004
    2 SQ=SQRT (1.+4.*A*E/U**2)                                          SVD  005
      P=.5*U/E                                                          SVD  006
      S=P*(1.+SQ)                                                       SVD  007
      V=P*(1.-SQ)                                                       SVD  008
      RETURN                                                            SVD  009
      END                                                               SVD  010
      FUNCTION YB(I,X)
      DIMENSION K1(4),AREA(1),KFN(1),RK(1),WU(1)
      READ(9'I)KFN,G1,I2,G2,G3,K1,A1,AREA,RK,A2,A3,A4,A5,WU
      MX=5
      KF=KFN(1)
      GO TO (13,23,33),KF
   13 YB=WU(1)/(AREA(1)*RK(1))
      YBB=YB
C     PRINT 300,I,KF,X,S(I),V(I),CC,YBB
  300 FORMAT(3H0YB, 2I10,5E20.4 )
      RETURN
   23 PFUN=0.0
      YB=0.0
      YBB=YB
C     PRINT 300,I,KF,X,S(I),V(I),CC,YBB
      RETURN
   33 WRITE(MX,100)I
  100 FORMAT('0 ERROR IN YB, I=',I8)
      RETURN
      END
      FUNCTION YDP(I,X,K)
      DIMENSION FAC(1),ZZ(1)
      READ(8'I)FAC,ZZ
      IF(K-1)1,1,2
    1 YDP=FAC(1)*PFUN(I,X,K,1.,1.)+ZZ(1)
      XXXXX=FAC(1)
      YYYYY=ZZ(1)
      GO TO 3
    2 YDP=FAC(1)*PFUN(I,X,K,1.,1.)
    3 RETURN
      END
```